Bayesian and Frequentist Regression Approaches for Very Large Data Sets

Leo Nikolaus Geppert

Fakultät Statistik

TU Dortmund

A thesis submitted for the degree of

doctor rerum naturalium (Dr. rer. nat.)

16^{th} of November 2018

Abstract

This thesis is concerned with the analysis of frequentist and Bayesian regression models for data sets with a very large number of observations. Such large data sets pose a challenge when conducting regression analysis, because of the memory required (mainly for frequentist regression models) and the running time of the analysis (mainly for Bayesian regression models). I present two different approaches that can be employed in this setting.

The first approach is based on random projections and reduces the number of observations to manageable level as a first step before the regression analysis. The reduced number of observations depends on the number of variables in the data set and the desired goodness of the approximation. It is, however, independent of the number of observations in the original data set, making it especially useful for very large data sets. Theoretical guarantees for Bayesian linear regression are presented, which extend known guarantees for the frequentist case. The fundamental theorem covers Bayesian linear regression with arbitrary normal distributions or non-informative uniform distributions as prior distributions. I evaluate how close the posterior distributions of the original model and the reduced data set are for this theoretically covered case as well as for extensions towards hierarchical models and models using q-generalised normal distributions as prior.

The second approach presents a transfer of the Merge & Reduce-principle from data structures to regression models. In Computer Science, Merge & Reduce is employed in order to enable the use of static data structures in a streaming setting. Here, I present three possibilities of employing Merge & Reduce directly on regression models. This enables sequential or parallel analysis of subsets of the data set. The partial results are then combined in a way that recovers the regression model on the full data set well. This approach is suitable for a wide range of regression models. I evaluate the performance on simulated and real world data sets using linear and Poisson regression models.

Both approaches are able to recover regression models on the original data set well. They thus offer scalable versions of frequentist or Bayesian regression analysis for linear regression as well as extensions to generalised linear models, hierarchical models, and q-generalised normal distributions as prior distribution. Application on data streams or in distributed settings is also possible. Both approaches can be combined with multiple algorithms for frequentist or Bayesian regression analysis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Et es wie et es, et kütt wie et kütt und et hätt noch immer jot jejange.

(Eetste Deil vum kölsche Jrundjesetz)

In the last years and decades, both available memory and computing power have increased by large factors. Together, this has lead to the possibility of storing as well as analysing very large data sets employing reasonably complex statistical models. However, the running time required to obtain results often grows linearly with the size of the data. This rate means that regression algorithms often do not scale well and make an analysis infeasible for very large data sets. In addition, while storing data sets typically does not pose a problem in terms of memory requirements, it may be necessary to load the data set into working memory, which exhibits considerably less storage capability.

In this thesis, I present two different approaches that make analysing regression models on very large data sets feasible. While my focus lies on Bayesian models, both of the approaches can be applied in a frequentist setting as well as in a Bayesian setting. Depending on the model at hand, the advantages that can be expected from the approaches are different—in the frequentist case, reducing the memory requirements is the most prominent advantage, while in the Bayesian case, the running time required often poses a more dire problem.

In the following, I will first introduce linear regression models in general, before introducing challenges that arise for the analysis of regression models on very large data sets.

1.1 Regression models

In this manuscript, I focus on regression techniques in both frequentist and Bayesian settings. I will introduce the basic concepts of regression analysis in the current section before presenting the challenges that arise when conducting regression analysis on very large data sets in Section 1.2.

Regression analysis is an established and well-studied statistical concept where the influence of p variables $\underline{x}_1, \ldots, \underline{x}_p, p \in \mathbb{N}$, on usually one variable $\underline{Y}$ is modelled and estimated. Because $\underline{Y}$ is modelled as depending on the values of $\underline{x}_1, \ldots, \underline{x}_p$, $\underline{Y}$ is called the dependent variable or response variable, while $\underline{x}_1, \ldots, \underline{x}_p$ are called the independent variables or explanatory variables. All of these vectors are elements of $\mathbb{R}^n$ where n is the number of observations. The independent variables can be seen as column vectors of a matrix $X \in \mathbb{R}^{n \times p}$. I refer to the row vectors of X as $\underline{X_{i\cdot}} \in \mathbb{R}^p, i = 1, \ldots, n$. They represent the i^{th} observation.

The most basic idea of connecting the two is via a linear model (see Model (1.1)). The matrix X and the vector $\underline{Y}$ are linked via the unknown vector $\underline{\beta}$ and an unobservable additive error term $\underline{\eta}$, resulting in the full model of

$$\underline{Y} = X\underline{\beta} + \underline{\eta}. \tag{1.1}$$

Following Groß (2003), assumptions (L1) – (L4) are made:

(L1) $X \in \mathbb{R}^{n \times p}$ is a non-stochastic matrix with full column rank p, where $n > p$.

(L2) $\underline{Y} \in \mathbb{R}^n$ consists of n observable random variables.

(L3) $\underline{\eta} \in \mathbb{R}^n$ consists of n non-observable random variables with $\mathsf{E}(\underline{\eta}) = \underline{0}_n$ and $\mathsf{Cov}(\underline{\eta}) = \sigma^2 I_n$.

(L4) $\underline{\eta} \overset{iid}{\sim} N(\underline{0}_n, \sigma^2 I_n)$,

where I_n is the identity matrix in $\mathbb{R}^{n \times n}$. (L1) – (L3) are strictly required for linear regression. (L4) is often added, but does not necessarily need to be fulfilled in order to employ linear regression. (L4) is, however, a requirement should one want to conduct significance tests on the elements of $\underline{\beta}$. If the assumption is met, $\underline{Y}$ also follows a normal distribution with expected value $X\underline{\beta}$ and variance-covariance matrix $\sigma^2 I_n$. For this reason, linear regression is usually conducted when $\underline{Y}$ follows a normal distribution. The matrix X can contain variables that are present in the data set, but it may also contain additional entries such as transformed variables, interactions between variables or an intercept term. For that reason, X is also called the design matrix.

In the thesis at hand, my focus lies on frequentist linear models as given in Model (1.1) or their Bayesian counterparts which are introduced in Section 1.1.2. In Chapter 3, approaches for handling large data sets in the context of two generalisations of the linear model are examined. In Chapter 4, a method is proposed that is able to handle both linear regression models and generalised linear regression models. The theoretical background of all extensions of the linear model will be presented in the respective chapters.

1.1.1 Frequentist regression analysis

Both linear models and generalised linear models can be applied in a frequentist setting as well as in a Bayesian setting. In a frequentist setting, each element of $\underline{\beta}$ is seen as an unknown fixed value. The aim is estimating $\underline{\beta}$ in a way that minimises a given error function. In the case of linear regression, this error function is given as

$$\min_{\underline{\beta}^*} \sum_{i=1}^{n} \left(y_i - \underline{X_{i\cdot}}\underline{\beta}^*\right)^2, \tag{1.2}$$

where $y_i,\ i = 1, \ldots, n$ are the observed values of the dependent variable and $\underline{X}_{i\cdot},\ i = 1, \ldots, n$ are the row vectors of X. The optimal solution $\hat{\underline{\beta}}$ of Equation (1.2) is called the ordinary least squares (OLS) estimator. It can be obtained via Equation (1.3):

$$\hat{\underline{\beta}} = \left(X'X\right)^{-1} X'\underline{Y}. \tag{1.3}$$

Under the assumptions (L1) – (L3), the OLS estimate is the best linear unbiased estimate with an expected value of $\mathsf{E}(\hat{\underline{\beta}}) = \underline{\beta}$ and variance of $\mathsf{Var}(\hat{\underline{\beta}}) = \sigma^2 \left(X'X\right)^{-1}$ (Groß, 2003). The estimator's variance-covariance matrix forms the basis for the estimated standard deviations, also called standard errors, which are given by

$$\mathrm{se}_{\hat{\beta}_j} = \sqrt{\sigma^2 \underline{e}_j' (X'X)^{-1} \underline{e}_j}, \quad j = 1, \ldots, p, \tag{1.4}$$

where $\underline{e}_j$ is the j^{th} unit vector, $j = 1, \ldots, p$. Additionally assuming (L4), the standard errors $\mathrm{se}_{\hat{\beta}_j}$ can be employed to conduct significance tests of the form $H_0 : \beta_j = 0$ vs. $H_1 : \beta_j \neq 0$. For linear regression, this is done using a t-test (Montgomery and Peck, 1992); for generalised linear models, Wald tests can be employed (Krämer and Sonnberger, 1986). In both cases, the ratio of estimate and standard error $\frac{\hat{\beta}_j}{\mathrm{se}_{\hat{\beta}_j}}$ forms the basis of the test.

Having obtained an estimate for the unknown parameter vector $\underline{\beta}$, checking the assumptions for possible violations as well as assessing the fit of the model is an important part of a regression analysis. In the following, I will present some basic approaches to that end. For further reading, especially on strengths and weaknesses of linear models, Groß (2003) and a multitude of literature from different areas of Statistics provide more details that are outside the scope of this thesis.

After conducting a regression analysis, it is necessary to check whether the assumptions are met and to run diagnostics in order to find out whether the model fit exhibits problems in general or for some values or areas of the domain of X. Residuals are one of the most important tools for model diagnostics in the regression context. They aid in assessing the goodness of fit of the regression model, may hint at systematic influences that are missing in the model and may alert when assumptions are violated. Two of the assumptions, (L3) and (L4) deal with the error term $\underline{\eta}$. As $\underline{\eta}$ cannot be observed, the residuals $\underline{r} = \underline{Y} - X\underline{\hat{\beta}}$ can be seen as estimated error term and serve as replacement. The raw residuals $\underline{r}$ are the difference of the observed values in $\underline{Y}$ and the values estimated by the model, $\underline{\hat{Y}} = X\underline{\hat{\beta}}$. (According to McCullagh and Nelder (1989), both Legendre and Gauß originally defined residuals the other way around: as difference of estimated and observed values.) The OLS estimator by definition has the property of minimising the sum of squared residuals amongst all other choices of $\underline{\hat{\beta}}$ (see Equation (1.2)).

Examining the raw residuals has some merits, but there is one important difference between the error term and the raw residuals: While the unknown error vector $\underline{\eta}$ is assumed to be homoscedastic and uncorrelated, this does not hold for the raw residuals, as $\underline{r} \sim \left(\underline{0}, \sigma^2 \left(I_n - H\right)\right)$, where $H = X(X'X)^{-1}X'$ is the hat matrix or projection matrix. This means that the variances $\mathsf{Var}(r_i)$ depend on i, which is not ideal for diagnostic purposes. For that reason, two further types of residuals are introduced. Following the terminology in Groß (2003), standardised residuals are defined as

$$\tilde{r}_i = \frac{r_i}{\hat{\sigma}\sqrt{(1 - h_{ii})}}, \quad i = 1, \ldots, n, \tag{1.5}$$

where h_{ii} is the i^{th} diagonal element of H. When replacing the estimated $\hat{\sigma}$ with the true, but unknown σ, it can easily be seen that $\mathsf{E}(\tilde{r}_i) = 0$ and $\mathsf{Var}(\tilde{r}_i) = 1$ for $i = 1, \ldots, n$.

Studentised residuals are closely related to standardised residuals. Here, instead of the general estimate of the variance σ^2, the variance is estimated separately for every observation by leaving the i^{th} observation out of the model. With this, studentised residuals are defined as

$$r_i^* = \frac{r_i}{\hat{\sigma}_{-i}\sqrt{(1-h_{ii})}}, \quad i = 1, \ldots, n, \tag{1.6}$$

where $\hat{\sigma}^2_{-i}$ is the estimated variance without observation i. Both standardised and studentised residuals are frequently in use.

One example are residual plots, which contain the fitted values according to the model, $\underline{\hat{Y}}$, on the x-axis and the standardised residuals $\underline{\tilde{r}}$ on the y-axis. If the model exhibits a good fit to the data and the assumptions are fulfilled, no structure can be found in the plot, i.e. the residuals are located around zero with constant variance. If, for example, all low residuals $r_i < 0$ are concentrated in a small area of the domain or the variance increases as $\underline{\hat{Y}}$ increases, this is an indication for problems with the model.

Studentised residuals can be and are also used instead of standardised residuals. Chatterjee and Hadi (1988) have shown that the relation between standardised and studentised is given by

$$r_i^* = \tilde{r}_i \sqrt{\frac{n-p-1}{n-p-\tilde{r}_i^2}}. \tag{1.7}$$

This means that standardised and studentised residuals are roughly equal when the residuals are in the interval $[-1, 1]$. For residuals with larger absolute value, the difference grows with studentised residuals exhibiting larger values in absolute terms. Thus, unusually high or low residuals become more conspicuous when using the studentised version. Studentised residuals can also be employed to construct tests as the r_i^* follow a t_{n-p-1} distribution (Chatterjee and Hadi, 1988).

In addition to residuals that analyse the distance between the observed value and the estimated value according to the model, it is also useful to examine how much influence the observations have on the parameter estimate and the variation of the estimate. In a setting with one explanatory variable and an intercept term, $\underline{Y} = \beta_0 \cdot \underline{\mathbb{1}}_n + \beta_1 \underline{X_{1\cdot}} + \underline{\eta}$, changes in the values of y_i have larger effects on the model for observations that are close to the extreme values of $\underline{y}$ and smaller effects for observations in the middle of the range of $\underline{y}$. To formalise the concept of differing importance, a number of measures have been introduced. Here, I concentrate on leverage scores and Cook's distance.

The concept of leverage is also based on the hat matrix H. The diagonal of H, h_{ii}, is not only employed to obtain standardised or studentised residuals, it also contains the so-called leverage scores $\underline{l}$, $l_i = h_{ii}, i = 1, \ldots, n$ (Groß, 2003). Leverage scores offer additional insight as they

concentrate on the importance of an observation for the model regardless of any discrepancy between expected and observed $\underline{Y}$-values.

All leverage scores lie between 0 and 1, $0 \leq l_i \leq 1, i = 1, \ldots, n$, where 1 stands for a very important observation that dominates the outcome of the model. To calculate leverage scores, only the design matrix X is needed, the values of $\underline{Y}$ are not taken into consideration. Leverage scores thus represent the potential influence the observations have. This means that removing an observation with a large leverage score does not necessarily have a big influence on the estimated regression coefficients. The observation in question may lie perfectly on the regression line or hyperplane of the model, i.e. the estimated coefficients may be very similar or even identical regardless of whether the observation in question is included in the model or not. However, it is also possible that the exclusion of the important observation leads to considerable changes in the model.

In contrast, Cook's distance (Cook, 1977; Groß, 2003; McCullagh and Nelder, 1989) also incorporates information about the $\underline{Y}$-values of the observations. Cook's distance is defined as

$$D_i = \frac{(\underline{\hat{\beta}} - \underline{\hat{\beta}_{-i}})' X' X (\underline{\hat{\beta}} - \underline{\hat{\beta}_{-i}})}{p\hat{\sigma}^2}, \quad i = 1, \ldots, n, \tag{1.8}$$

where $\underline{\hat{\beta}_{-i}}$ is the estimate for $\underline{\beta}$ without $\underline{X_{i\cdot}}$, the i^{th} observation. Equation (1.8) is equivalent to (cf. Groß, 2003; McCullagh and Nelder, 1989)

$$D_i = \frac{l_i \tilde{r}_i^2}{p(1 - l_i)}, \quad i = 1, \ldots, n. \tag{1.9}$$

Equations (1.8) and especially (1.9) make it clear that Cook's distance combines leverage scores l_i with standardised residuals $\tilde{r}_i$. High values of Cook's distance mean that observations with a high potential influence also exhibit large residuals. Removing such observations from the model would lead to a large change in the estimated values of $\underline{\beta}$.

1.1.2 Bayesian regression analysis

In a Bayesian setting, $\underline{\beta}$ itself is seen as a random variable with a density. In addition to the likelihood, which fulfils the same role as Model (1.1), we need to specify a prior distribution $p(\underline{\beta})$, which represents the knowledge we have about $\underline{\beta}$ beforehand. The prior knowledge may vary from hardly any or none to a lot of information from earlier research or studies. This knowledge can be reflected in the choice of $p(\underline{\beta})$. Analogously, prior knowledge about the error term variance σ^2 is taken into the model. Model (1.10) shows a Bayesian linear regression model

with unspecified prior distributions. The likelihood $\mathcal{L}(\underline{Y}|X\underline{\beta})$ is identical to the likelihood in Model (1.1). Prior distributions for the parameters $\underline{\beta}$ and σ^2 are then added, resulting in

$$\begin{aligned} p(\underline{\beta}) &\sim \text{prior distribution} \\ p(\sigma^2) &\sim \text{prior distribution} \\ \mathcal{L}(\underline{Y}|X\underline{\beta}) &\sim N(X\underline{\beta}, \sigma^2) \\ p(\underline{\beta}, \sigma^2|X, \underline{Y}) &\propto L(\underline{Y}|X\underline{\beta})p(\underline{\beta})p(\sigma^2). \end{aligned} \tag{1.10}$$

In a Bayesian regression analysis, the aim is to draw conclusions about the posterior distribution, which is given by $p(\underline{\beta}, \sigma^2|X, \underline{Y})$ for a linear regression model. The posterior distribution is a compromise between the information in the data (i.e. the likelihood) and the information that was known beforehand (i.e. the prior distribution). In Model (1.10), the posterior distribution is given as being proportional to the product of likelihood and prior distribution. This product needs to be divided by a normalising constant in order to represent a distribution. Gelman et al. (2014) offer a general introduction to Bayesian statistics as well as ways of choosing prior distributions.

When the posterior distribution and the prior distribution stem from the same family, the prior is called a conjugate prior and the model is called a conjugate model (Gelman et al., 2014, chapter 2). When the prior is conjugate, the posterior is available as closed-form expression and can be obtained analytically. Such cases allow for easy and fast Bayesian analysis. Well-known conjugate models are the Normal-Normal model where both the prior distribution and the likelihood are modelled as normal distributions (in case of the likelihood with known variance). The posterior distribution consequently is a normal distribution as well. In a regression context, modelling the prior distribution for $\underline{\beta}$ with a normal distribution and the prior distribution for σ^2 with an inverse Gamma distribution also results in a conjugate model. Other examples are the Beta-Binomial model where the likelihood is modelled as binomial distribution while prior and posterior distribution follow Beta distributions as well as the Poisson-Gamma model.

While conjugate models are very useful and allow for easy analysis, they are not available for many models. In such cases, an analytical solution usually cannot be obtained. This is due to the normalising constant, which is in fact a high-dimensional integral that often is not analytically solvable. Instead, Laplace approximation or Markov Chain Monte Carlo (MCMC) techniques can be used. Both are active areas of research. In Appendix A, an overview of approaches formerly and currently used in the analysis of Bayesian models is presented. The notation is

kept to that of regression models, but the algorithms introduced can be applied to a broad range of models analogously.

In addition to the references given in Appendix A, Bolstad (2010) offers an overview of computational methods employed to obtain an approximation of the posterior distribution.

In the thesis at hand, only Hamiltonian Monte Carlo (HMC) is employed. This is a variant of MCMC that is widely used in practice. The R-package `rstan` (Stan Development Team, 2018) offers an easy way of conducting Bayesian analyses with HMC in R.

After successfully obtaining an MCMC-sample, often descriptive or non-parametric methods are chosen to analyse it. Values like mean, median, variance, inter-quartile range, and quantiles can be employed to characterise the univariate marginal posterior distributions. Boxplots and kernel density estimates offer a good visual representation. 95% credible intervals can be obtained from the 2.5% and 97.5% quantiles. In a regression context, they allow an evaluation of the importance of a variable, similarly to a significance test in a frequentist setting. It is also possible to contemplate two or multiple marginal distributions simultaneously. However, this is usually not the focus of a regression analysis.

1.2 Challenges for regression analysis for Big Data

According to Moore's law (Mack, 2011), computing power – represented by the number of components in a typical semiconductor integrated circuit – doubles every 12-18 months. The law has empirically been true since the 1970's, but seems to level off in recent years. The size of data sets has also been growing at a fast rate over the last years, especially since digital ways of saving data have become readily available (Hilbert and López, 2011). In the last five years, the term Big Data has reached the public. While the advent of large data sets (in comparison to the computing power available) may not be a recent development, this development poses a challenge to statistical methods and their scalability. Many statistical concepts are based on sample sizes of n in the double or triple digits. One prominent representative is the p-value, which loses some usability for large data sets as even tiny differences between means, say, easily return results that are deemed statistically significant.

Growing data sets pose a challenge when conducting regression analysis, both on a computing and on a methodological level. The memory required grows at least linearly with the dimensions n and p. This may be a problem in both frequentist and Bayesian cases as standard regression analysis procedures depend on the whole data set being available.

The running time for standard procedures typically depends linearly on n and p as well. In the frequentist setting, this poses less of a problem as the result is forthcoming after one pass through the data. In a Bayesian setting, most algorithms require multiple passes through the data, which means that considerable running time is necessary before results become available. For the effects of this on a practical example, please refer to Figure 2.1 in Chapter 2. In both cases, linear regression analysis may become infeasible or even impossible for very large data sets, but in a Bayesian setting the problem is more apparent.

In response to these challenges, Welling et al. (2014) suggest two desiderata methods should fulfil in order to be suitable for Bayesian analyses on very large data sets or Big Data. The first desideratum is that each update only accesses a subset of the data set where the size of the subset does not depend on the size of the data set, i.e. is independent of n. This requirement makes methods feasible for streaming data which can be interpreted as having infinitely large data sets, thus making sure that the method works regardless of how big the data set grows to be.

As a second requirement, Welling et al. (2014) postulate that the algorithm should be adequate for an application in distributed systems. This is useful as very large data sets may be distributed to different computers for analysis as a means of lessening the demands for every single machine or to suit the architecture of current computer clusters. It also opens up the possibility of performing the analysis in cyber-physical systems or generally on a distributed system with comparatively low computing power.

In addition to suggestions concerning the computational demands of the methods, there are also methodological issues to consider. One prominent example is the concept of statistical significance. As a general rule, test statistics depend on the number of observations n – in the example of a one-sample t-test the dependence is $\sqrt{n}$. This leads to larger values of the test statistic as n increases. While this is a generally desirable property, it can lead to rejection of the null hypothesis even for very small deviations of the estimated value from the tested value. This may lead to significant results for effects that are not relevant in practice.

1.3 Outline of the thesis

In the following, I will present different ideas regarding how to conduct regression techniques as data sets grow. Throughout the thesis, the number of observations n is considered large while the number of variables p is small to moderate.

Chapter 2 introduces a possibility of reducing data sets with large n and small to medium-sized p for frequentist linear regression as well as Bayesian linear regression with normal or vague prior distributions. The results on the reduced data set are provably close to the ones obtained on the original data set. Here, the focus lies on a reduction of the size of the data set by reducing the number of observations from n to k. This leads to a decrease in the memory required to conduct the analysis. When employing MCMC-methods or other algorithms where the running time may be long and depends on n, this also leads to a considerable reduction in the total running time of the analysis.

With Chapter 2 as foundation, Chapter 3 generalises the results to more regression models by allowing a broader choice of prior distributions. This is done by including hierarchical models, which introduce hierarchical information on an additional level, and p-generalised normal distributions. These can be used for penalised regression approaches like LASSO- regression and ridge regression.

Chapter 4 introduces the method "Merge & Reduce" that was originally developed in the context of clustering in very large data sets and transfers this method to the regression context. Ideas are presented for linear and Poisson regression models. Merge & Reduce splits the data set into blocks that are analysed iteratively. The information from each block is aggregated in an efficient way, eventually resulting in a final model that represents the original model. This approach does not reduce the data set, but by splitting the data set it offers a possibility to handle very large data sets or data streams in an efficient and stable way.

Chapter 5 summarises and discusses the results. Limitations of the proposed methods and possible future directions of work are also picked up on. In addition, some results of Bachelor's and Master's theses that were developed within project C4 of SFB 876 are introduced and reviewed.

MCMC-methods are employed in most chapters of this thesis. They are not in the focus, but form the basis for most of the empirical evaluations. For that reason, Appendix A introduces MCMC-methods and their idea. Appendix B investigates the effects of violations from the assumptions of regression models as well as observations of varying importance. This presents an interesting issue, because all methods that reduce the number of observations work well if all observations are of the same importance for the model and all assumptions are met.

Chapter 2

Random Projections for Linear Regression

Wo die beherrschende Idee des Studentenlebens Amt und Beruf ist, kann sie nicht Wissenschaft sein. Sie kann nicht mehr in der Widmung an eine Erkenntnis bestehen, von der zu fürchten ist, daß sie vom Wege der bügerlichen Sicherheit abführt.

(Walter Benjamin – Das Leben der Studenten)

2.1 Introduction to random projections

The idea behind random projections is to find a random matrix Π that reduces the dimension of a vector from n to k such that for any vector $\underline{v} \in \mathbb{R}^n$, there exists a random matrix $\Pi \in \mathbb{R}^{k \times n}$ for which Equation (2.1) holds:

$$(1 - \varepsilon)\|\underline{v}\| \leq \|\Pi\underline{v}\| \leq (1 + \varepsilon)\|\underline{v}\|, \tag{2.1}$$

where the so-called approximation parameter ε influences how close the projection is to the original vector. The idea was first formalised in the Johnson-Lindenstrauss theorem (Johnson and Lindenstrauss, 1984). Since then, there has been active research in this area, with one focus on finding algorithms that create suitable random matrices Π.

Random projections are employed to reduce the larger dimension. In the following, I assume $n \gg p$ and that the aim is a reduction of n to k, where $n > k > p$. Random projections can also be used to reduce the number of variables in the case of $n < p$, however, an interpretation of the new variables is not easily possible. Section 5.1.1 provides more details and thoughts on this. A

reduction below the rank of X is possible, but comes at the cost of a potential catastrophic loss, i.e. an unacceptably high approximation error. Random projections are also called sketches. In the following, I will especially refer to the process as sketching and to the reduced data set as sketched data set.

One field of application of the Johnson-Lindenstrauss theorem is linear regression. In the frequentist case, the quadratic loss function

$$\min_{\underline{\beta}^*} \sum_{i=1}^{n} \left(y_i - \underline{X_{i.}}\underline{\beta}^*\right)^2 = \arg\min_{\underline{\beta}^*} \|\underline{Y} - \underline{X_{i.}}\underline{\beta}^*\|_2^2$$

on which the estimate for $\underline{\beta}$ is based, can be interpreted as a vector. This way, the values of the loss function for different choices of $\underline{\beta}$ and also the optimal solution $\hat{\underline{\beta}}$ are approximately recovered. How to construct suitable random matrices Π is an active area of research. Important results from this research as well as related approaches are discussed in Section 2.2.

A desirable property for random projections and other techniques that reduce the size of the data is that the goodness-of-approximation is controllable employing the parameter ε. To that end, Definition 1 introduces a formal concept of ε-subspace embeddings.

Definition 1 (ε-subspace embedding)**.** *Given a matrix $U \in \mathbb{R}^{n\times p}$ with orthonormal columns, an integer $k \leq n$ and an approximation parameter $0 < \varepsilon \leq 1/2$, an ε-subspace embedding for U is a function $\Pi : \mathbb{R}^n \to \mathbb{R}^k$ such that*

$$(1-\varepsilon)\|Ux\|^2 \leq \|\Pi Ux\|^2 \leq (1+\varepsilon)\|Ux\|^2 \tag{2.2}$$

holds for all $x \in \mathbb{R}^p$, or, equivalently

$$\|U^T\Pi^T\Pi U - I_p\| \leq \varepsilon. \tag{2.3}$$

In Definition 1, the spectral norm of a matrix U is defined as follows:

Definition 2 (spectral norm)**.** *The spectral or operator norm of a matrix $U \in \mathbb{R}^{n\times p}$ is defined as*

$$\|A\| = \sup_{x\in\mathbb{R}^p\setminus\{0\}} \frac{\|Ax\|}{\|x\|},$$

where $\|x\| = (\sum_{i=1}^n x_i^2)^{\frac{1}{2}}$ denotes the Euclidean vector norm for any $x \in \mathbb{R}^n$.

In Section 2.3 I introduce three sketching methods that can be used to obtain ε-subspace embeddings. Other sketching methods as well as other techniques can also be used to arrive at ε-subspace embeddings. I will discuss some of them in Section 2.2.

2.2 Related work

To give an overview of related research in the fields of Statistics and Computer Science, I will pick up on two main fields: reduction of the dimension of the original data set in a linear regression context (both frequentist and Bayesian) as well as making the MCMC-algorithms employed for Bayesian analysis more efficient. The efficiency of algorithms employed for frequentist regression analysis usually poses a negligible problem and is not considered in this section.

Size of the data set Reducing the dimensionality of the data set is a common aim and has been widely used in both Statistics and Computer Science. Usually, the focus is on reducing the number of variables p. One example is principal component analysis (PCA) (Jolliffe, 2002), which originally had the main aim of avoiding problems with multicollinearity by replacing the original variables with an orthogonal basis. PCA can also be and is used for the purpose of dimensionality reduction. This is possible, because each principal component represents a decreasing amount of the total variation present in the data set. Only employing the first p^* principal components can thus reduce the dimensionality while keeping the main structure in the data.

In the thesis at hand, the focus lies on reducing the number of observations from n to k, where $n \gg p$ and $n > k > p$. This is especially useful in the context of very large data sets, because the running time of commonly used algorithms for statistical analysis depends on n. In this situation, tried and tested techniques based on PCA include principal component regression and partial least squares (Friedman et al., 2009). More recent results that are based on PCA can be found in the field of coresets for k-means clustering. Such approaches aim at finding a small subset that approximates the original data set with an approximation error of, say, $(1 \pm \varepsilon)$ with respect to the objective function (Feldman et al., 2013). This means that the value of the objective function at the optimum found on the subset lies within the interval $[(1-\varepsilon)v_{opt}, (1+\varepsilon)v_{opt}]$, where v_{opt} is the value of the objective function at the optimum found on the original data set. Here, the idea is to combine iterative simple random sampling without replacement and the additional exclusion of observations that are close to already sampled observations. A related concept, data squashing, attempts to compress the data based on the likelihood of the observations (DuMouchel et al., 1999; Madigan et al., 2002). The main objective is to preserve statistical information in such a

way that statistical analyses carried out on the squashed data set approximate the results on the original data set well. To that end, the data set is partitioned using likelihood-based clustering. Then, pseudo-observations are created from these clusters and employed in further statistical analyses.

Random projections have been studied in Computer Science to achieve low-rank approximation (Cohen et al., 2015a), for least squares regression (Clarkson and Woodruff, 2009; Sarlós, 2006), in the context of Gaussian process regression (Banerjee et al., 2013), for clustering problems (Boutsidis et al., 2010; Cohen et al., 2015a; Kerber and Raghvendra, 2014) and classification tasks (Paul et al., 2014) as well as for compressed sensing (Candès et al., 2006; Donoho, 2006). Baraniuk et al. (2007) employ random projections to approximate a family of subspaces with only sparse vectors. In compressed sensing, Bayesian inference has been proposed for efficient computation by Ji and Carin (2007).

Recently, random projections and other algorithms using randomised linear algebra have been studied with a focus on statistical aspects. Raskutti and Mahoney (2015) as well as Ma et al. (2014) study subsampling approaches based on leverage scores and their statistical properties extensively. Ma et al. (2014) show that leverage scores behave uniformly for data that is generated in such a way that it follows linear regression models. Yang et al. (2015) employ several sketching and subsampling methods as a fast means to precondition the data set before efficiently finding the least squares estimator. They also discuss algorithms that work in parallel and distributed and support their results with extensive empirical evaluations on very large data sets. Our work continues investigating statistical properties of random projections by transferring these algorithms to the Bayesian setting.

Bayesian regression analysis for very large data sets has been considered before in Statistics. In the case of "large p, small n", Guhaniyogi and Dunson (2015) proposed reducing the number of variables using random projections. Under several assumptions they show that the approximation converges to the desired posterior distribution. For the general case, this is not possible as dimensionality reductions that are done oblivious to the target variable can lead to additive error in the worst case (Boutsidis and Magdon-Ismail, 2014).

In the "large n, small p"-case, Tall Skinny QR (TSQR) (Demmel et al., 2012) is a QR-decomposition that requires multiple passes over the data, but can be easily calculated in parallel (Benson et al., 2013; Constantine and Gleich, 2011). TSQR can be used as a preprocessing step prior to Bayesian inference using MCMC-methods. It leads to a stable decomposition with high accuracy. However, it is restricted to regression models with a normally distributed likelihood

and requires the data to be given row-by-row. Due to the QR-decomposition being expensive, the running time exhibits a lower bound of $\Omega(np^2)$ (cf. Demmel et al., 2012). $\Omega(\cdot)$ stands for the Bachmann-Landau-notation and indicates that a function is asymptotically bounded below by the given function. In this chapter as well as in Chapter 4, I also employ $O(\cdot)$, which signifies that a function is asymptotically bounded above by the function given.

In a recent publication, Giraldi et al. (2018) combine random projections with ideas from information theory to arrive at "optimal projections". These are based on the Kullback-Leibler divergence and the mutual information between the projected observations and the parameters of interest.

In this thesis, I only consider so-called data-oblivious subspace embeddings. Such subspace embeddings are constructed independently of the data set that is to be embedded, i.e. Π in Definition 1 is independent of U. This allows for efficiently obtaining Π. In contrast, their counterparts – data-aware subspace embeddings – are adapted according to the data set, i.e. Π depends on U. Data-aware subspace embeddings are also useful in large data settings, but are outside the scope of this thesis.

Efficiency of the algorithm Markov Chain Monte Carlo methods provide the gold standard for Bayesian analysis whenever no conjugate model that is analytically solvable is available. MCMC-methods are reliable and offer insights into possible problems with the model or with the approximation of the posterior distribution. However, the required running time depends on $n \cdot p$ and can be quite high, constituting a bottleneck.

There is a rich field of attempts to improve this situation. One line of research is the improvement of MCMC-algorithms, e.g. Hamiltonian Monte Carlo methods (confer Appendix A). The advantage of easily being able to judge whether the result is an approximation of the desired posterior distribution is preserved. Other approaches are to tweak or replace the MCMC-algorithm. One cause for the bottleneck is the repeated evaluation of the likelihood, which usually involves all n observations.

Balakrishnan and Madigan (2006) propose an algorithm that reads the data block-wise and performs a number of MCMC-steps on the block. For every new block, some of the data points are kept while others are replaced according to weights that try to capture the importance of the observations for the current batch of data. The algorithm only requires one pass through the data, but provides the user with a full MCMC-sample. The selection rule employed is justified

empirically. Theoretical support is only present for the univariate case and is based on central limit theorems for Sequential Monte Carlo methods.

An approach by Bardenet et al. (2014) suggests subsampling the data in order to approximate the decision to accept or reject in every step of a Metropolis-Hastings algorithm. They show that the approximated decision is similar to the original one with high probability in every iteration. The number of iterations in the subsampling algorithm is not fixed beforehand, but calculated according to an adaptive stopping rule. This means that the number of iterations depends on the variance of the logarithm of the likelihood ratios considered.

Two recent articles by Quiroz et al. (2018a,b) suggest replacing the Metropolis-Hastings and the MCMC-algorithm, respectively, by a version that accepts or rejects the proposed point based on a subsample of the data set. This reduces the computational cost of the algorithm, thus providing a more efficient way of obtaining the posterior distribution in a Bayesian setting. They propose inclusion probabilities proportional to the observations' contribution to the likelihood, which is approximated by a Gaussian Process. For Metropolis-Hastings (Quiroz et al., 2018b), the suggested algorithm includes a delayed acceptance mechanism, which aims at only evaluating the likelihood when there is a good chance of acceptance. In both cases, there may be some relation to data-squashing mentioned above, however, Quiroz et al. (2018a) aim at providing an alternative to classical MCMC-type algorithms. There is a lack of theoretical approximation guarantees, but the articles indicate useful applicability empirically.

There are also approaches that replace the MCMC-algorithm completely. One of them is the Integrated Nested Laplace Approximation (INLA) (Rue et al., 2009). Here, instead of Monte Carlo approximation of the posterior distribution, Laplace approximation is employed. This type of approximation is only suitable for the class of latent Gaussian models. However, this class offers a great number of available models (Martins et al., 2013) and INLA is generally faster than MCMC for models from this class. One disadvantage is that it is harder to verify how good the approximation of the posterior distribution is for a concrete case.

Approximate Bayesian Computation (ABC) offers an alternative to Bayesian analysis when it is not possible to calculate the likelihood. Instead, simulations approximating the likelihood function are employed (Csilléry et al., 2010). This approximation is not quite comparable to others mentioned in this overview, as the main aim is not a more efficient analysis, but rather making a Bayesian analysis possible where otherwise it would be impossible. ABC can provide summary statistics for very low dimensions, as a general rule of thumb $p < 10$ (Beaumont et al., 2002; Csilléry et al., 2010).

After considering related work from different angles and perspectives, I will now introduce our own proposition, starting with our theoretical results. While the theoretical guarantees are of interest for the thesis at hand, they are not the focus of my work and are consequently only presented to a certain extent. Please refer to Geppert et al. (2017) and Munteanu (2018) for the full extent of theoretical analyses.

2.3 Theoretical results

In a Bayesian regression analysis, the aim is analysing the posterior distribution $p(\underline{\beta}|X, \underline{Y})$. A scheme of our approach is given in Sketch (2.4):

$$\begin{array}{ccc} [X, \underline{Y}] & \xrightarrow{\Pi} & [\Pi X, \Pi \underline{Y}] \\ \downarrow & & \downarrow \\ p(\underline{\beta}|X, \underline{Y}) & \approx_{\varepsilon} & p(\underline{\beta}|\Pi X, \Pi \underline{Y}). \end{array} \tag{2.4}$$

The left-hand side of Sketch (2.4) depicts the normal procedure of a Bayesian regression analysis. Starting with X and $\underline{Y}$, we sample from the posterior distribution to obtain $p(\underline{\beta}|X, \underline{Y})$. However, for large data sets, this path may take long or even be unfeasible. When employing random projections, the first step is to embed the data set into a lower dimension with the help of the random matrix Π. All subsequent analyses will then be performed on the reduced data set, on the right-hand side of Sketch (2.4). This greatly reduces the requirements with regard to running time and memory.

Importantly, results on the reduced data set need to be close to results on the original data set, indicated by $\approx_{\varepsilon}$ in Sketch (2.4). In the frequentist case, finding the distance between two estimates for $\underline{\beta}$ is easy. In contrast, we are dealing with the distance between two posterior distributions in the Bayesian case. Here, the Wasserstein distance is one of the suitable possibilities. Let γ and ν be two probability measures over $\mathbb{R}^p$ and $\Lambda(\gamma, \nu)$ denote the set of all joint probability measures over $\mathbb{R}^p \times \mathbb{R}^p$ that have γ and ν as marginal distribution. The ℓ_2-Wasserstein distance is defined as follows:

Definition 3 (Wasserstein distance, following Villani (2009))**.** *Given two probability measures γ, ν on $\mathbb{R}^p$ the ℓ_2-Wasserstein distance between γ and ν is defined as*

$$\mathcal{W}_2(\gamma, \nu) = \left(\inf_{\lambda \in \Lambda(\gamma,\nu)} \int_{\mathbb{R}^p \times \mathbb{R}^p} \|\underline{x} - \underline{y}\|_2^2 \ \mathrm{d}\lambda(\underline{x}, \underline{y}) \right)^{\frac{1}{2}} = \inf_{\lambda \in \Lambda(\gamma,\nu)} \mathsf{E}_{\lambda} \|\underline{x} - \underline{y}\|_2^{2\,\frac{1}{2}}.$$

In Definition 3, the Wasserstein distance is defined for the case of ℓ_2. Other norms can analogously be employed. The special case of $q = 1$, i.e. the ℓ_1-Wasserstein distance is also known as Earth Mover's distance. In the thesis at hand, I employ the ℓ_2-Wasserstein distance to evaluate the distance between the two posterior distributions, based on the original data set and based on the sketched data set.

I consider three ways of obtaining random projections that have been proven to be useful for frequentist linear regression and transfer the results to Bayesian linear regression. The three methods are the Rademacher matrix (RAD), the Subsampled Randomised Hadamard Transform (SRHT) and the Clarkson-Woodruff-method (CW). The methods differ in certain aspects that will be introduced in the following as well as in Table 2.1 on page 22. The sketches from all three sketching methods are linear. This means that instead of sketching the data set in one go, sketching subsets of the data set and aggregating the resulting sketches by simply adding them leads to the same approximative result. This is especially useful when batches of data are analysed sequentially, e.g. in a streaming setting, or when the sketching is done in parallel on multiple computers.

Every random projection may fail to provide the theoretically guaranteed goodness-of-approximation. This happens with a probability of δ which directly influences the target number of observations k – the so-called failure probability. Two things are important to note here. Firstly, the word fail is used in a rather weak sense. It does indicate that the guaranteed goodness-of-approximation could not be held by the random projection, but usually, the approximation will only be slightly worse. It is not indicative of a catastrophic failure by any means. Catastrophic failures may happen, but the event of a meteor destroying the computer that calculates the embedded data set can be seen as more likely.

Secondly, it is easily possible to find out whether the approximation works as intended or not. To that end, the practitioner is encouraged to sketch the original data set twice and compare the frequentist regression models based on both sketches. If the resulting estimates of $\underline{\beta}$ are not similar, a failure is likely to have occurred. During the analyses of simulated and real data sets for the thesis at hand and for Geppert et al. (2017), I have not come across one instance of a failed random projection. In this section, the failure probability δ is included whenever appropriate. However, δ will not be explicitly mentioned when dealing with experimental results in further sections.

Rademacher Matrix (RAD) The Rademacher matrix offers a very simple form of the sketching matrix $\Pi \in \mathbb{R}^{k \times n}$. Each cell is chosen independently from $\{-1, 1\}$ with equal probability. The matrix is then rescaled by multiplying every entry with $\frac{1}{\sqrt{k}}$, giving Π. This is also the oldest of the three methods considered in this thesis and in Geppert et al. (2017). Sarlós (2006) showed that this method forms an ε-subspace embedding when choosing $k = O\left(\frac{p \log(p/\delta)}{\varepsilon^2}\right)$, where δ is the failure probability. The required number of observations was later reduced to $k = O\left(\frac{p+\log(1/\delta)}{\varepsilon^2}\right)$ in Clarkson and Woodruff (2009), which is the lower bound (as proven by Nelson and Nguyễn, 2014). Employing the RAD method yields a very good reduction of the data set. In our implementation, RAD and SRHT, which is introduced shortly, lead to equally small sketches. However, the RAD embedding requires a running time of the order $\Theta(npk)$, which is considerably more than the two other embedding methods presented here. The running time can be reduced when the input is given row by row or block by block due to a fast matrix multiplication algorithm employed in our implementation.

As a technical remark, it is not necessary to sample the entries of D from $\{-1, 1\}$ in a way that full row-wise independence of the RAD-matrix is given. Instead, as noted first by Alon et al. (1999), four-wise independence is provably sufficient, i.e. any four or less entries of an arbitrary row behave as if they were fully independent. This enables efficient sampling of random numbers via a hashing scheme that generates BCH-codes, which are a well-known class of codes in Computer Science, employing a seed of size $O(\log n)$. Our implementation is based upon the four-wise independent BCH-scheme described in Rusu and Dobra (2007).

Subsampled Randomised Hadamard Transform (SRHT) The Subsampled Randomised Hadamard Transform consists of three components, $\Pi = \frac{1}{\sqrt{k}} R H_m D$. $R \in \mathbb{R}^{k \times m}$ is a row sampling matrix, which means that exactly one entry per row is a 1 while all others are 0. The position of the cells containing the 1-entry is sampled independently from $\{1, \ldots, m\}$ with equal probability for all elements of the set. m is the smallest power of two that is larger than n, i.e. $\exists i$ such that $2^i = m$ where $m \geq n$ and $2^{(i-1)} < n$. Using m instead of n allows more efficient calculation of the second component H_m, the Hadamard matrix of order m. Hadamard matrices are matrices with special properties: they are square matrices whose entries are either 1 or -1 and whose rows are mutually orthogonal. The third component, $D \in \mathbb{R}^{m \times m}$, is a diagonal matrix where each element of the diagonal is sampled from $\{-1, 1\}$ with equal probability. As a final step, the matrix is rescaled by $\frac{1}{\sqrt{k}}$ as before.

Even though in practice $m > n$, this does not pose a problem. We can think of the design matrix X being padded with additional 0-entries to obtain a size of $m \times p$. These additional entries remain 0 after multiplying ΠX, which means that this does not need to be done explicitly. For the regression analysis we just employ the first n columns or rows, as appropriate.

This sketch was originally proposed by Ailon and Liberty (2009). Its target dimension is $k = O\left(\frac{(\sqrt{p}+\sqrt{\ln n})^2 \ln(p/\delta)}{\varepsilon^2}\right)$ (Boutsidis and Gittens, 2013). The small dependency on n is negligible if $n = O(\exp(p))$. In practice, this is fulfilled when p is reasonably large. In that case, a target dimension of $k = \Omega(p \ln p)$ is necessary. The running time required to obtain an SRHT-sketch is $O(np \ln k)$, which is considerably less than for a RAD-sketch. According to Ailon and Liberty (2009), the construction of SRHT-sketches is closely related to four-wise independent BCH-codes. Although, to the best of our knowledge, there is no explicit proof, we again employ the four-wise BCH-scheme to obtain D. From the empirical evaluations, this seems to work well in practice.

Clarkson-Woodruff (CW)-sketch The Clarkson-Woodruff-sketch (Clarkson and Woodruff, 2013) is the most recent method considered in the thesis at hand as well as in Geppert et al. (2017). Here, $\Pi = \frac{1}{\sqrt{k}}\Phi D$. The rescaling factor is the same as for RAD and SRHT. $D \in \mathbb{R}^{n \times n}$ again is a diagonal matrix whose entries are sampled from $\{-1, 1\}$ with equal probability. $\Phi \in \mathbb{R}^{k \times n}$ is a matrix with 0 and 1 as entries. The positions of the 1-entries are determined according to a random map $h : \{1, \ldots, n\} \to \{1, \ldots, k\}$. For every $i \in \{1, \ldots, n\}$, the image $h(i)$ is sampled from $\{1, \ldots, k\}$ with equal probability $\frac{1}{k}$ for every element. The entries $\Phi_{h(i),i}, i = 1, \ldots, n$, are set to 1, all other elements of Φ are set to 0.

Due to its construction, the CW-sketch can be applied to any $(n \times p)$-matrix in $O(np)$ time, which in the case of sparsity in X reduces to the number of non-zero entries. Concerning the running time required, this is optimal up to small constants. Reading a data set or data stream and sketching it using the CW-sketch only adds a small multiplicative factor. For empirical results on this, please refer to Section 2.4.7. The target dimension is $k = \Omega(p^2)$ (Nelson and Nguyễn, 2013b), which is a higher dependency on p in comparison to the other two sketching methods. Nelson and Nguyễn (2013a) improved the upper bounds given by Clarkson and Woodruff (2013) and showed that with probability $1 - \delta$ a target dimension of $k = O(\frac{p^2}{\varepsilon^2 \delta})$ is sufficient to provide an ε-subspace embedding. Following Nelson and Nguyễn (2013a) further, it is sufficient to use four-wise independent random entries to generate D. To that end, we employ the BCH-scheme by Rusu and Dobra (2007) in our implementation.

The three methods differ in the running time they require to sketch the data set as well as in the target dimension k of the sketched data for a given value of ε. Table 2.1 provides an overview. Even though theoretically, the target dimension k is different for RAD and SRHT, both sketching methods result in the same target dimension for any value of ε in our implementation. However, they do differ in the running time required. CW is the fastest of the three methods. Its running time depends linearly on the number of non-zero entries in the data set. The size of the resulting sketch has a higher dependence on the number of variables p than the other two methods, as k is of the order p^2. For small values of p, CW will return sketches with a smaller target dimension, but for values of around $p = 60$, this changes. In general, CW provides a very fast and reliable sketch, but k may become too large if p is large.

The choice of the sketching method represents a trade-off between speed and size. However, this trade-off is more relevant for a frequentist linear regression. For Bayesian models, the size of the data set greatly influences the running time of the MCMC-algorithm while the running time required to obtain the sketch is negligible. If a low total running time of the analysis is a priority, employing a sketching method that results in a small target dimension k may be of more importance.

For all three sketching methods, the target dimension is independent of n. They are also easily adequate for an application in distributed systems with the only caveat that for SRHT, care must be taken to ensure that n is always larger than k. All three thus fulfil both desiderata by Welling et al. (2014) mentioned in Section 1.2.

Whether or not the sketches are able to handle cases where $k > n$ can be an important difference. This situation can occur in a streaming situation when incoming data are analysed in batches or in a parallelised setting when the data set is distributed onto different machines for sketching purposes. Because k is independent of n, the sketch may be larger than the original data set for relatively small n. Of the three methods introduced here, RAD and CW are able to handle such situations whereas SRHT is not.

For all three methods, we theoretically show in Geppert et al. (2017) that the results on the embedded data sets are close to the results on the respective original data sets. How close the results are is controlled by the parameter ε. Larger values of ε lead to smaller size of the sketched data set but also to larger approximation errors. Conversely, lower values of ε mean that the approximation is close to the original results, but this comes at the cost of less reduction in the size of the data set.

sketching method	target dimension	running time	can handle $k > n$
RAD	$O\left(\frac{p+\ln(1/\delta)}{\varepsilon^2}\right)$	$O(npk)$	yes
SRHT	$O\left(\frac{p\cdot\ln(p/\delta)}{\varepsilon^2}\right)$	$O(np\ln k)$	no
CW	$O\left(\frac{p^2}{\varepsilon^2\delta}\right)$	$O(\mathrm{nnz}(X)) = O(np)$	yes

Table 2.1: Comparison of the three considered ε-subspace embeddings, giving the target dimension k depending on the approximation parameter as a function of ε and the running time required to obtain the ε-subspace embedding. $\mathrm{nnz}(X)$ denotes the number of non-zero entries in X, δ denotes the failure probability.

Especially interesting are Lemma 4 and Theorem 1 from Geppert et al. (2017). Both are concerned with theoretical guarantees for the ℓ_2-Wasserstein distance between the posterior distribution on the original data set $p = p(\underline{\beta}|X, \underline{Y})$ and the posterior distribution on the sketched data set $p' = p(\underline{\beta}|\Pi X, \Pi\underline{Y})$. Lemma 4, given here as Equation (2.5), states that the distance between p and p' is at most

$$\mathcal{W}_2^2(p, p') \leq \frac{\varepsilon^2}{\mathrm{sv}_{min}^2} \|X\underline{\hat{\beta}} - \underline{Y}\|_2^2 + \varepsilon^2 \operatorname{tr}\left((X'X)^{-1}\right), \tag{2.5}$$

where $\underline{\hat{\beta}}$ is the estimator for $\underline{\beta}$ in a frequentist sense or the posterior mean for $\underline{\beta}$ in a Bayesian sense. sv_{min} is the smallest singular value resulting from a singular value decomposition of X. Theorem 1 from Geppert et al. (2017) is very similar in its structure. While Equation (2.5) only embeds the likelihood, Theorem 1 also includes the prior distribution, which has to be either an arbitrary normal distribution or a non-informative vague prior over the whole of $\mathbb{R}^p$.

Equation (2.5) and analogously Theorem 1 from Geppert et al. (2017) state that the difference between both posterior distributions depends only on an ε-fraction of the model's goodness-of-fit plus an ε-fraction of the variation in the data, represented by the trace of $(X'X)^{-1}$. This makes the role of the approximation parameter ε clear.

In our R-package `RaProR`[1], it is also possible to set the desired target value k directly instead of choosing ε. Here, a lower value of k corresponds with a larger approximation error and vice versa. The choice of ε or k thus represents another trade-off in the Bayesian case. If the focus lies on obtaining a result quickly, larger values of ε lead to a higher reduction and a faster subsequent analysis at the cost of a higher approximation error. If more time is available, a lower value of ε ensures a better approximation.

[1] http://ls2-www.cs.tu-dortmund.de/grav/de/projekte/RaProR

2.4 Simulation study

2.4.1 Data generation and models

In addition to the theoretical considerations, I also conduct an extensive simulation study to investigate the benefits of the method in practice. Table 2.2 contains a synopsis of the parameters used to create different simulated data sets. The dimensions of the data set, n and p theoretically play different roles. The number of variables p has a large influence on the reduced number of observations k while k is independent of the number of observations n. To corroborate this empirically, both n and p are included as parameters. In general, every combination of settings is repeated several times with different underlying parameter vectors $\underline{\beta}$.

According to Equation (2.5), the goodness of approximation depends on the goodness of fit of the model. For that reason, I include the error term variance σ^2 as a parameter. A linear regression model is suitable for all data sets created in this simulation study, however, while the unexplained variance is small for some data sets, there are others with a medium or high error term variance.

parameter	role	values
n	number of observations	$\{50\,000, 100\,000, 500\,000, 1\,000\,000\}$
p	number of variables	$\{50, 100\}$
σ	error term standard deviation	$\{1, 2, 5, 10\}$

Table 2.2: Overview of parameters in simulation study.

The "true" underlying values of $\underline{\beta}$ are generated employing a zero-inflated Poisson distribution. More precisely, the components of $\underline{\beta}$ are set to 0 with probability 0.5 and follow a Poisson distribution with rate $\lambda = 3$ with probability 0.5. All components are multiplied with (-1) with probability of 0.5, allowing for negative influences in the regression model. To obtain the data set X, we first draw the column means $\underline{\bar{X}}_{\cdot j}$ from a $N(0, 25)$ distribution. The values in every column are drawn from a $N(\underline{\bar{X}}_{\cdot j}, 4)$ distribution. A 1-column is added to model the intercept term. In the last step, the values of $\underline{\eta}$ are drawn from $N(0, \sigma^2)$ and $\underline{Y}$ is calculated as $\underline{Y} = X\underline{\beta} + \eta$.

The Bayesian regression model consists of a standard likelihood, i.e. $\underline{Y}$ is assumed to follow a normal distribution,

$$\underline{Y} \sim N(X\underline{\beta}, \sigma^2 I_n). \tag{2.6}$$

As prior distribution, we employ improper uniform distributions over $\mathbb{R}$ for all components of $\underline{\beta}$ and an improper uniform distribution over $\mathbb{R}^+$ for σ. The posterior distribution is thus proportional to

$$p(\underline{Y}|X\underline{\beta}) \propto L(\underline{\beta}|X, \underline{Y}) \cdot \mathbb{1}_p. \tag{2.7}$$

This results in a proper posterior distribution, provided, $L(\underline{\beta}|X, \underline{Y})$ contains enough information, which does not pose a problem in a large data setting. For this model, closed-form expressions can be calculated. However, this simulation study forms the basis for subsequent simulation studies with more complicated models that cannot be solved analytically. For that reason, I compare and present results obtained from the package `rstan` for this chapter as well.

The simulations in this thesis are conducted using R (R Core Team, 2018) while the Bayesian analyses are carried out using the R-package `rstan`(Stan Development Team, 2018). The sketches are calculated with our R-package `RaProR`. The simulations are conducted on an Intel Xeon E5430 quad-core CPU running at 2.66 GHz using 16 GB DDR2 memory on a Debian GNU Linux 7.8 distribution. The hard drive used is a Seagate Momentus 7200.4 G-Force 500 GB, SATA 3 GB/s HDD with 7200 rpm and 16 MB cache. The Bayesian regression models are obtained via the function `stan` from the package `rstan`. In my settings, the function employs the No-U-Turn-Sampler (NUTS)(Hoffman and Gelman, 2014). I use the standard settings of four chains, which run in parallel.

2.4.2 Overview of the results

In the simulation study, I create a total of 32 data sets, one for each of the combinations in Table 2.2. I conduct a Bayesian analysis on every data set to obtain the results on the original data set. However, this was only successful for parts of the data sets due to resource limitations.

For every data set, I construct a sketch for each of the three methods RAD, SRHT, and CW described in Section 2.3. I set the parameter ε to values of 0.1 and 0.2, resulting in a total of six sketches per simulated data set. I compare the results of the Bayesian analyses on the sketches with the true values of $\underline{\beta}$ and where possible with the results on the original data sets.

2.4.3 Comparison of running times

In a first step, I conduct a preliminary simulation study aimed at observing the running time of a Bayesian linear regression model when the number of observations n grows. Figure 2.1 contains the resulting running times for several values of $n \in [50, 50\,000]$ with p fixed at $p = 52$ parameters

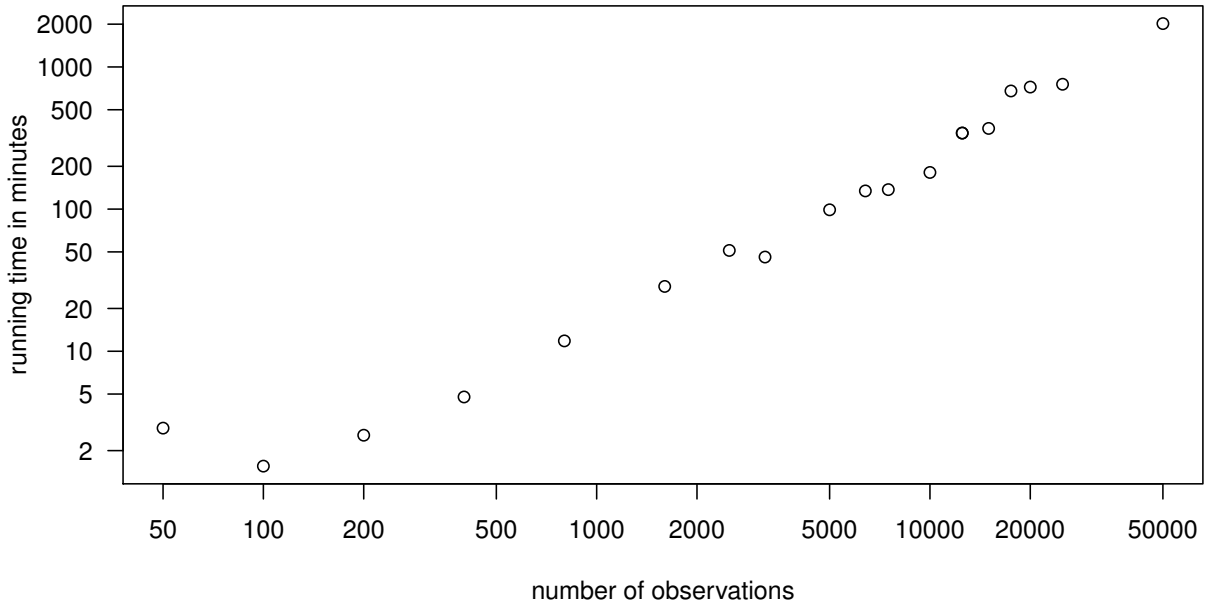

Figure 2.1: Running times for simulated data sets with varying number of observations and $p = 52$ variables.

(50 independent variables plus the intercept plus the variance term). The running time for the analyses seems to depend linearly on the number of observations with occasional jumps. The value for $n = 50$ seems to be an outlier, this might be caused by the number of variables being larger than the number of observations. The linear dependency on n does not pose a problem for small to medium-sized data sets, but for Big Data settings, only methods whose dependency on the growing dimension is sublinear scale well and remain feasible, confer e.g. desiderata in Muthukrishnan (2005). This underlines the usefulness of sketched data sets.

Recall that the target dimension k is independent of n. From the theory, we expect that doubling the number of observations only influences the running time required to obtain the embedded data set, while subsequent analyses are not affected as the two embedded data sets are of the same size. In the present case, where the total running time is dominated by the subsequent analysis, doubling the number of observations is expected to have a very small effect.

Table 2.3 contains the target dimension k of the sketches depending on the number of variables and the value of the approximation parameters ε.

Table 2.4 gives an overview of the running times grouped into the categories "Preprocessing" and "Analysis". The latter group measures the running time required to obtain a converged Bayesian linear regression model on the respective data set. The "Preprocessing"-group differs for original and embedded data sets. For original data sets, the value stands for the running time required to read the data set and load it into memory. For embedded data sets, the value

p	ε	RAD	SRHT	CW
52	0.1	20 547	20 547	16 384
52	0.2	5 137	5 137	4 096
102	0.1	47 175	47 175	65 536
102	0.2	11 794	11 794	16 384

Table 2.3: Target dimension k for the three sketching methods sketches for different values of p and ε

represents the time needed to create the embedded data set. The total running time required to obtain a Bayesian regression model on a sketched data set is thus given by adding the running times for reading the data, sketching it and then analysing the sketch.

Figure 2.2 exemplarily shows the effect of doubling the number of observations on the running time. For the original data set, the given running times consist of reading and analysing the data; for the three sketching methods, the times also include the time required for sketching. When doubling the number of observations in the data set from 50 000 to 100 000, the total running time required for the original data set more than doubles from just over 600 to more than 2000 minutes. In contrast, the running times for the three sketches show no clear trend. There is an increase for RAD and CW, but a decrease for SRHT. While the time required for the sketching process doubles for the three methods (confer Table 2.4), the running times for the analysis only show some slight variations that seem to be caused by chance. As the running times for the analysis are orders of magnitude larger than the running times for reading and sketching, there is no systematic effect visible.

This is not the most efficient way of handling large data sets. The running times required for reading the data sets are based on the basic execution of the command `read.table`. The package `data.table` (Dowle and Srinivasan, 2017) offers a more efficient option for reading in large data sets in the function `fread`. Both functions can also be employed in such a way that they iteratively read parts of the data set instead of the whole data set at once, thus reducing the required memory considerably. This is especially advantageous when sketching data sets. Due to the linearity of the sketches, it is possible to obtain the same sketch as on the whole data set by iteratively sketching parts of the data and simply adding these sketches. Our R-package `RaProR` offers such functionality, but for the sake of comparison, we decided to employ the standard version of reading the data sets.

The impact of this is low when the number of observations is relatively low (here: $n \leq 100\,000$) as the total running time is dominated by the time required for the Bayesian analysis. I will

			Preprocessing				Analysis			
n	sketch	ε	$\sigma = 1$	$\sigma = 2$	$\sigma = 5$	$\sigma = 10$	$\sigma = 1$	$\sigma = 2$	$\sigma = 5$	$\sigma = 10$
$5 \cdot 10^4$	none		0.32	0.41	0.43	0.44	1096	749.1	616.5	498.7
$5 \cdot 10^4$	RAD	0.1	1.60	1.68	1.68	1.73	315.1	213.4	156.8	154.7
$5 \cdot 10^4$	SRHT	0.1	0.03	0.02	0.03	0.03	317.3	278.4	181.9	166.4
$5 \cdot 10^4$	CW	0.1	0.01	0.01	0.01	0.01	375.2	293.9	164.6	171.8
$5 \cdot 10^4$	RAD	0.2	0.40	0.39	0.42	0.43	23.17	26.00	17.48	21.81
$5 \cdot 10^4$	SRHT	0.2	0.02	0.02	0.02	0.02	29.04	30.65	23.26	26.00
$5 \cdot 10^4$	CW	0.2	0.01	0.01	0.01	0.01	26.92	25.77	20.57	22.94
$1 \cdot 10^5$	none		0.69	0.83	1.02	1.05			2036	1617
$1 \cdot 10^5$	RAD	0.1	3.27	3.41	3.41	3.27	278.9	260.80	167.2	182.9
$1 \cdot 10^5$	SRHT	0.1	0.05	0.06	0.05	0.05	285.0	282.20	128.6	196.5
$1 \cdot 10^5$	CW	0.1	0.02	0.03	0.02	0.02	257.5	278.20	187.0	198.8
$1 \cdot 10^5$	RAD	0.2	0.76	0.84	0.84	0.80	21.44	23.21	17.52	23.65
$1 \cdot 10^5$	SRHT	0.2	0.04	0.04	0.05	0.04	23.72	26.82	21.52	22.70
$1 \cdot 10^5$	CW	0.2	0.02	0.02	0.02	0.02	21.94	26.29	21.22	23.45
$5 \cdot 10^5$	none		5.49	5.16	5.92	5.71				
$5 \cdot 10^5$	RAD	0.1	16.88	15.96	16.10	16.36	279.8	313.3	165.9	198.4
$5 \cdot 10^5$	SRHT	0.1	0.20	0.21	0.20	0.21	310.2	308.0	190.8	190.2
$5 \cdot 10^5$	CW	0.1	0.09	0.09	0.09	0.10	335.7	300.3	189.3	166.9
$5 \cdot 10^5$	RAD	0.2	3.73	4.00	4.03	3.85	27.37	27.19	17.22	19.58
$5 \cdot 10^5$	SRHT	0.2	0.19	0.18	0.19	0.19	31.37	25.62	22.26	24.76
$5 \cdot 10^5$	CW	0.2	0.09	0.08	0.09	0.09	26.03	25.23	24.39	22.86
$1 \cdot 10^6$	none		18.23	12.88	12.59	14.09				
$1 \cdot 10^6$	RAD	0.1	51.77	147.4	33.75	34.71	209.2	279.0	215.8	145.6
$1 \cdot 10^6$	SRHT	0.1	0.41	0.49	0.61	0.62	341.1	265.0	294.0	154.8
$1 \cdot 10^6$	CW	0.1	0.19	0.27	0.38	0.46	281.7	232.4	175.5	144.0
$1 \cdot 10^6$	RAD	0.2	7.92	8.46	8.38	8.21	21.27	19.93	22.87	23.43
$1 \cdot 10^6$	SRHT	0.2	0.39	1.44	0.68	0.68	26.61	31.32	19.69	23.69
$1 \cdot 10^6$	CW	0.2	0.21	0.19	0.45	0.39	28.58	19.50	22.05	9.72

Table 2.4: Running times for data sets with $p = 52$. Columns 4 to 7 ("Preprocessing") contain the running times of the sketching methods in minutes, for the original data set, the values represent the time required to read the data set into memory, which is a prerequisite for every sketching method. The four columns to the right ("Analysis") contain the running times of the Bayesian linear regression analysis in minutes.

first concentrate on these smaller data sets and the comparison of the total running times on the original data sets with those on the sketched data sets. After that, I will comment on trends that can be observed as n grows.

2.4.4 Comparison of posterior means

The first comparison is akin to a frequentist analysis. To evaluate the results, I compare the means of the posteriors $p(\underline{\beta}|X, \underline{Y})$ on the sketched data sets with those on the respective original

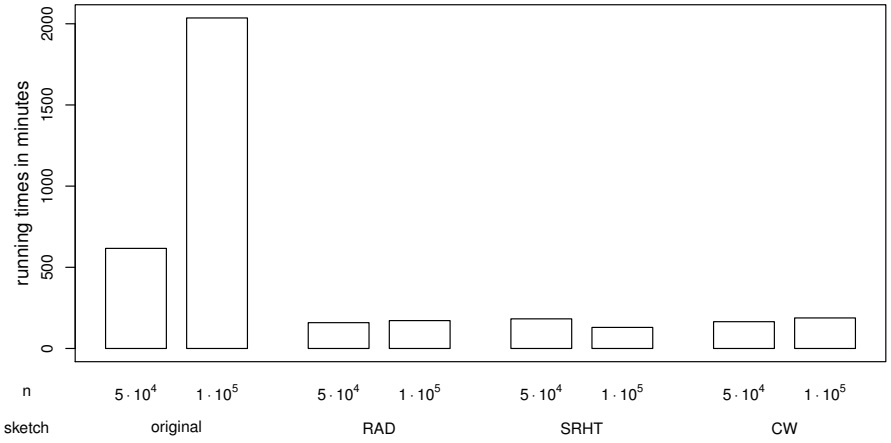

Figure 2.2: Total running times in minutes for data sets with $n \in \{50\,000, 100\,000\}$, $p = 52$, $\sigma = 5$ and approximation parameter $\varepsilon = 0.1$. For the sketched data sets, the total running time consists of the time for reading, sketching and analysing the data set. For the original data set, the sketching time is 0 as this step is not applied.

n	sketch	ε	$\sigma = 1$	$\sigma = 2$	$\sigma = 5$	$\sigma = 10$
$5 \cdot 10^4$	RAD	0.1	0.052	0.025	0.021	0.834
$5 \cdot 10^4$	RAD	0.2	0.014	0.781	0.892	1.512
$5 \cdot 10^4$	SRHT	0.1	0.001	0.009	0.021	0.165
$5 \cdot 10^4$	SRHT	0.2	0.004	0.077	0.093	0.757
$5 \cdot 10^4$	CW	0.1	0.025	0.004	0.021	0.195
$5 \cdot 10^4$	CW	0.2	0.016	0.040	0.156	0.915
$1 \cdot 10^5$	RAD	0.1			0.836	0.958
$1 \cdot 10^5$	RAD	0.2			0.061	0.777
$1 \cdot 10^5$	SRHT	0.1			0.025	0.964
$1 \cdot 10^5$	SRHT	0.2			0.171	0.617
$1 \cdot 10^5$	CW	0.1			0.056	3.844
$1 \cdot 10^5$	CW	0.2			2.624	2.937

Table 2.5: Squared Euclidean distances between posterior mean values of the original model and models based on the respective sketches.

data sets where possible. For all cases, I compare the posteriors on the sketched data sets with the true values of $\underline{\beta}$.

Table 2.5 contains the sum of squared distances between the results on the embedded data sets and the original data sets. Geometrically, this is the squared Euclidean distance between the posterior mean vectors. It corresponds to the comparison of different $\hat{\underline{\beta}}$ in a frequentist setting. As indicated by Equation (2.5), the squared Euclidean distance grows with the standard deviation of the error term. There does not seem to be a systematic difference in performance between the different sketching methods. With larger ε, we usually, but not necessarily, observe

n	sketch	ε	$\sigma = 1$	$\sigma = 2$	$\sigma = 5$	$\sigma = 10$
$5 \cdot 10^4$	none		0.000	0.003	0.065	4.614
$5 \cdot 10^4$	RAD	0.1	0.048	0.016	0.124	1.718
$5 \cdot 10^4$	RAD	0.2	0.012	0.710	0.506	10.845
$5 \cdot 10^4$	SRHT	0.1	0.001	0.018	0.032	3.372
$5 \cdot 10^4$	SRHT	0.2	0.005	0.059	0.046	8.721
$5 \cdot 10^4$	CW	0.1	0.022	0.011	0.046	6.474
$5 \cdot 10^4$	CW	0.2	0.014	0.056	0.089	1.870
$1 \cdot 10^5$	none				0.065	0.035
$1 \cdot 10^5$	RAD	0.1	0.007	0.031	1.354	0.679
$1 \cdot 10^5$	RAD	0.2	0.033	0.009	0.117	0.579
$1 \cdot 10^5$	SRHT	0.1	0.030	0.136	0.040	0.696
$1 \cdot 10^5$	SRHT	0.2	0.007	0.125	0.387	0.453
$1 \cdot 10^5$	CW	0.1	0.004	0.232	0.022	4.496
$1 \cdot 10^5$	CW	0.2	0.011	0.072	3.484	3.473
$5 \cdot 10^5$	RAD	0.1	0.009	0.223	0.563	12.920
$5 \cdot 10^5$	RAD	0.2	0.045	0.322	1.729	0.658
$5 \cdot 10^5$	SRHT	0.1	0.009	0.147	0.418	0.059
$5 \cdot 10^5$	SRHT	0.2	0.016	0.033	0.085	2.978
$5 \cdot 10^5$	CW	0.1	0.027	0.097	1.305	0.153
$5 \cdot 10^5$	CW	0.2	0.050	0.009	0.135	3.579
$1 \cdot 10^6$	RAD	0.1	0.001	0.016	0.126	3.967
$1 \cdot 10^6$	RAD	0.2	0.080	0.011	0.072	1.357
$1 \cdot 10^6$	SRHT	0.1	0.002	0.010	0.599	0.288
$1 \cdot 10^6$	SRHT	0.2	0.002	0.183	2.029	4.329
$1 \cdot 10^6$	CW	0.1	0.002	0.289	1.202	4.445
$1 \cdot 10^6$	CW	0.2	0.003	0.047	0.100	0.395

Table 2.6: Squared Euclidean distances between true mean values and posterior means of models based on the respective sketches.

an increase in the distance. Please note that some values are missing, because the original models did not converge within reasonable time bounds.

In addition to the comparison to the original models' mean, I also compare the posterior means to the true means. Table 2.6 contains the squared Euclidean distances between the true mean for $p = 52$ and varying values of σ. The general picture looks very similar to the results in Table 2.5. The original model often exhibits the smallest squared Euclidean distances, but sometimes models based on embedded data sets are closer to the true mean. Again, there does not seem to be a systematic difference between the sketching methods. The squared Euclidean distances do not seem to be influenced by the value of n, with some squared Euclidean distances even exhibiting smaller values for larger n.

2.4.5 Comparison of fitted values

After this comparison on the level of parameters – whose number is not changed by sketching – I will compare the models on the level of observations, of which the sketches contain merely a fraction of the number of observations in the original data set. I multiply X with the posterior mean vector of $\underline{\beta}$, where this posterior mean can be based on the original data set or on the respective sketches. In a frequentist sense, these are fitted values $\underline{\hat{Y}}$, but all X-values are taken from the original data set, not necessarily from the data set the model is based on. This is done to see how close the approximation is on the level of $\underline{Y}$-values for both $\varepsilon = 0.1$ and $\varepsilon = 0.2$. Figure 2.3 is a scatter plot which shows two-dimensional kernel density estimates of the observations. The observations themselves are only included for sparsely populated areas of the plot to avoid overplotting. The fitted values based on the original model are on the x-axis while the fitted values based on the CW-sketch (with $\varepsilon = 0.1$) are on the y-axis. Darker shades of black stand for more observations. Even though the fitted values are based on one of the data sets with the highest standard deviation of the error ($n = 50\,000, \sigma = 10$), all values are close or reasonably close to the bisecting line. This means that the fitted values obtained by the two models do not differ by much. To get a better overview, Figure 2.4 depicts the distances between the fitted values as box plots. Here, all three sketching methods with both $\varepsilon = 0.1$ and $\varepsilon = 0.2$ are included. All six sets of distances are centred around zero. The effect of the approximation parameter ε is evident from the boxplot (2.4), the variation is larger for $\varepsilon = 0.2$ regardless of the sketching method. When fixing ε, all three sketching methods exhibit similar results, although the RAD-sketch seems to introduce slightly more variation into the differences than the other two sketching methods.

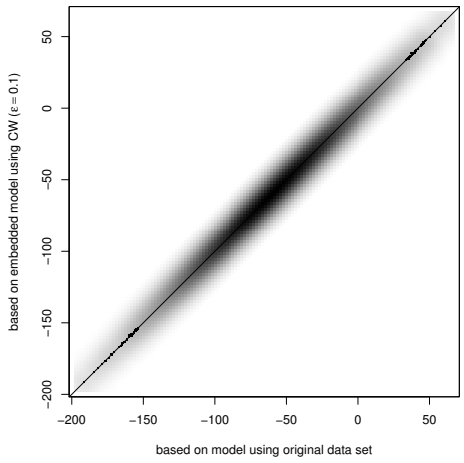

Figure 2.3: Comparison of fitted values based on the original data set with $n = 50\,000$, $p = 52$, $\sigma = 10$ and a CW-sketch with $\varepsilon = 0.1$. Darker shades of black stand for more observations.

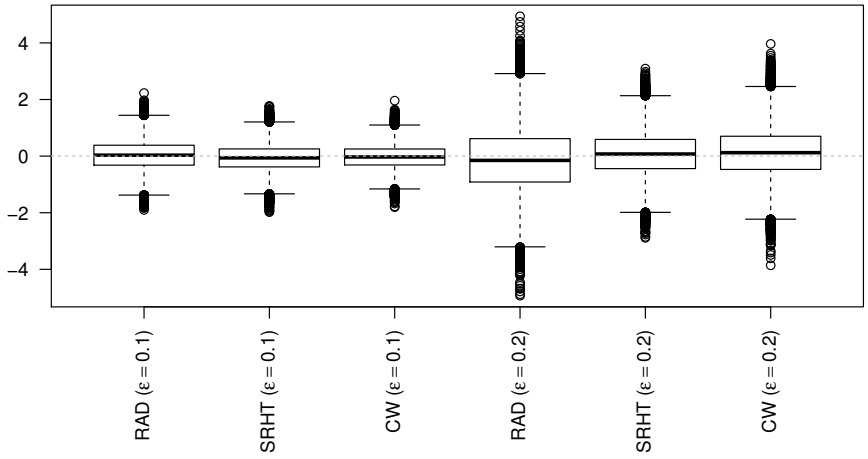

Figure 2.4: Difference of fitted values according to models based on the respective sketching methods and fitted values according to model based on original data set with $n = 50\,000$, $p = 52$, $\sigma = 10$.

These results mean that the posterior mean of $\underline{\beta}$ is recovered well enough by the three sketching methods to allow for prediction $\underline{\hat{Y}}$ on the original data set with only small deviations for most of the observations. For new data, the posterior predictive distribution is appropriate. Compared to the posterior distribution, it adds extra variation to reflect the additional uncertainty that stems from forecasting unknown data. How the additional variation from the posterior predictive distribution reacts with the additional variation introduced by sketching the data, would be an interesting question.

2.4.6 Comparison of posterior distributions

As I have conducted Bayesian regression the model consists not only of a mean value, but of a whole posterior distribution for each parameter. Figure 2.5 contains two exemplary boxplots of MCMC-samples representing marginal posterior distributions. The original data set contains $n = 50\,000$ observations, $p = 52$ variables and has an error standard deviation of $\sigma = 5$. The medians of the MCMC-samples based on the original data set are well-represented by the MCMC-samples based on sketches. Even though the median based on an embedded data set might be higher or lower for certain parameters, I did not find any systematic biases. The embedding introduces additional variation, which depends on the value of the approximation parameter ε, but does not seem to be influenced by the choice of sketching method.

In regression, a common task is the identification of important variables by means of variable selection. In a Bayesian setting, this can be done via credible intervals, among other approaches. The results indicate that the identification of important variables is quite accurately possible

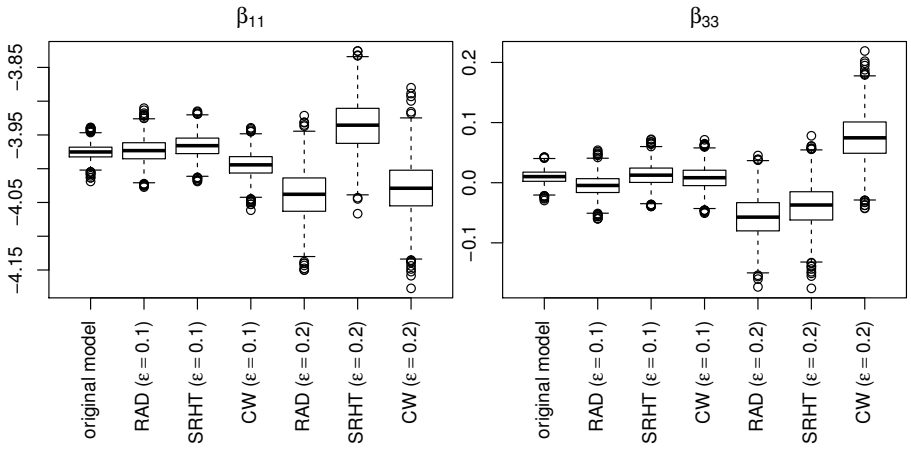

Figure 2.5: Boxplots of MCMC-sample for two parameters based on data set with $n = 50\,000$, $p = 52$, $\sigma = 5$ and respective sketches.

based on the resulting approximate models. However, one has to take the additional variation into account. Exemplarily, when using 95% credible intervals as criterion, one should not compare the endpoints of the credible interval to a fixed value μ. Instead, take the extra variation in the posterior distribution and also possible small shifts of the mean and median into account. For that reason I recommend using smaller values of ε when aiming at variable selection (see Figure 2.5).

2.4.7 Streaming experiment and remarks

Most data sets in the simulation study so far as well as the real data example (see Section 2.5) are of a size that enables a Bayesian analysis on the original data set within reasonable time. This enables comparisons of the results on the original and the sketched data sets, but it makes analysis of very large data sets unfeasible. As conclusion of the simulation study, we now consider an example that can be called Big Data. To that end, we generate and at the same time sketch a data set following the same rules described in Section 2.4.1 with an error term standard deviation of $\sigma = 0.1$. The original data set has a dimensionality of $10^9 \times 100$, where every entry is a double precision value. In CSV-format, this equates to around 2 TB and in binary representation, at least 750 GB of memory are required.

We employ the CW sketching method and obtain an embedded data set of dimensionality $65\,536 \times 100$. In CSV format, the sketched data set required around 140 MB and easily fits into the working memory. On this sketched data set, we conduct a Bayesian regression analysis in 2781 minutes. In this example, we cannot compare the result on the sketch to the result on

the original data set, but the squared Euclidean distance between the true values of $\underline{\beta}$ and the posterior mean of the Bayesian analysis calculates to $3.741 \cdot 10^{-6}$.

As mentioned in Section 2.3, reading in the data and calculating the sketch only adds a small multiplicative factor to the required running time compared to only reading the data in for CW-sketches. In our simulation study, we found the factor to lie between 1.01 and 1.04. Typically, higher factors are observed for small data sets and lower factors for large data sets (compare Table 2.4).

In some cases the algebraic structure might strongly depend on a few observations or variables. In such situations it is important to identify these or to retain their contribution in the reduced data set. So far, our model assumptions did not suffer from such ill-behaved situations, but in Geppert et al. (2017) we assess the performance of our method in this case. We construct data sets where an important part of the target variable falls into a subspace that is spanned by a small constant number of observations. Uniform random subsampling will pick these only with probability $O(\frac{1}{n})$. Oblivious subsampling techniques in general will have trouble identifying the important observations. In contrast, oblivious subspace embedding techniques preserve these effects with high probability. This effect is observed in practice even when comparing one sketch against the best of 1000 repetitions of uniform random subsampling.

2.5 Bike rental data set

In addition to the simulation study presented in Chapter 2.4, I apply the technique of random projections to a data set containing the number of rented bikes per hour in Washington, D.C., USA. The data set is taken from Fanaee-T and Gama (2014) via the UCI Machine Learning Repository[1] (Dheeru and Karra Taniskidou, 2017). It consists of $n = 17\,379$ hours of observation.

The data set contains the number of rental bike users per hour, divided into the categories "registered users" and "casual users". In my model, I only use the total number of users. This number is given as count data, but there are around 850 distinct values. For that reason, I treat it as continuous variable. After a square-root transformation, the variable shows some signs of bi-modality, but fits reasonably well to a normal distribution to justify conducting linear regression.

As explanatory variables I use the variables given in Table 2.7. Three of the variables – apparent temperature, humidity, and windspeed – are standardised in the data set to fall into

[1] http://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset

the interval $[0, 1]$. The other variables are factor variables. There are some variables in the data set that I do not consider in the model, because they are highly correlated with variables already included in the model. One example are the variables temperature and apparent temperature. Including both would be disadvantageous or even critical from a modelling point of view.

Originally, the variable *weathersit* has four levels ("clear", "cloudy", "light rain", and "heavy rain"). The last category only appears three times in the whole of the data set. To avoid problems with such a rare occurrence, I combine levels 3 and 4 to obtain the new level 3, which can be interpreted as "rainy weather". The other factor variables do not exhibit any problems with the frequency of levels.

Variable	Description	Remark
cnt	number of rental bikes used	dependent variable
season	season of the year	factor (4 levels)
yr	year (2011 or 2012)	factor (2 levels)
hour	hour (0 to 23)	factor (24 levels)
holiday	public holiday	factor (2 levels)
weekday	day of the week	factor (7 levels)
weathersit	weather situation ("clear" to "rain")	factor (3 levels)
atemp	apparent temperature	standardized
hum	humidity	standardized
windspeed	windspeed	standardized

Table 2.7: Variables from the bike sharing data set used in the model.

To handle the factor variables correctly, I first use the R function `model.matrix` which creates the correct design matrix X, and pass this matrix to `rstan` and `RaProR`, respectively. With all dummy variables and the intercept, the design matrix is of dimension $17\,379 \times 40$. Because of the relatively small number of observations in comparison to the number of variables, using $\varepsilon = 0.1$ as in the simulation study would result in an embedded matrix that is larger than the original design matrix. Instead, I here choose $\varepsilon = 0.15$ and $\varepsilon = 0.2$ for both the RAD- and the SRHT-sketches. This results in 6 767 and 3 807 observations in the sketched data sets. For the CW-sketches, I choose values of k that are closest to the target dimension of the other sketches, i.e. $k = 8192$ and $k = 4096$, respectively. Table 2.8 gives an overview of the sketch sizes given the sketching method and ε.

I will first present the results of the model on the original data set before comparing them to the results on reduced data sets. All variables included in the model show explanatory value for the number of bike rentals. Many of them show the direction one would expect beforehand. The apparent temperature has a high positive effect on the number of bike rentals while wind

p	ε	RAD	SRHT	CW
40	0.15	6 767	6 767	8 192
40	0.20	3 807	3 807	4 096

Table 2.8: Number of observations of the sketches for the bike sharing example. Different values of ε are used for RAD- and SRHT-sketches; the target dimension of CW-sketches is chosen to be the power of two closest to the size of the RAD- and SRHT-sketches.

speed and humidity exhibit a negative effect, but to a lesser extent than the effect of temperature. Heavy clouds or overcast conditions reduce the number of rental bikes compared to sunny weather or only light clouds, but the negative effect of rain in comparison to sunny weather is a lot higher.

Winter is the least popular season for rental bike users. Perhaps surprisingly, autumn has the highest positive effect of all seasons followed by spring and then summer. This is mainly due to the higher apparent temperatures in summer, which are already taken into account by the variable *atemp*.

There is also a distinct hourly effect. Compared to the hour between midnight and 1 a.m., the nightly hours are characterised by a low demand in rental bikes with a low point in the hour from 4 a.m. to 5 a.m. For the rest of the day, the number of rental bike users is higher compared to the baseline. There are two peaks, in the morning from 8 a.m. to 9 a.m. and in the afternoon from 5 p.m to 6 p.m. and 6 p.m. to 7 p.m.

The days of the week seem to have a smaller influence in general. The highest positive effects can be found for Friday and Saturday, the lowest for the baseline Sunday, followed by Monday. Compared to these effects, the variable *holiday*, which indicates whether a day was a public holiday or not, has a relatively large negative effect. This possibly indicates that a substantial amount of the rentals come from commuters, although there seems to be some additional demand by leisure time users on Fridays and Saturdays.

Finally, the variable *yr* exhibits a positive effect for the year 2012 compared to 2011, which indicates that the service generally attracted more rentals in the second year under consideration here.

All of the conclusions are similar on the sketched data sets. To check how close the results on the sketched data sets are compared to the results on the original data set, I employ a similar approach to Section 2.4. Table 2.9 contains the squared Euclidean distances between the posterior means of the original model and the models based on the sketched data sets. For the RAD-sketch, the difference between $\varepsilon = 0.15$ and $\varepsilon = 0.2$ is surprisingly high, for the other two sketching methods, the squared Euclidean differences do not increase as much when ε

ε	RAD	SRHT	CW
0.15	1.790	2.349	0.907
0.2	6.511	2.732	1.657

Table 2.9: Sum of squared distances between posterior mean values of the original model and models based on the respective sketches for the bike sharing data set.

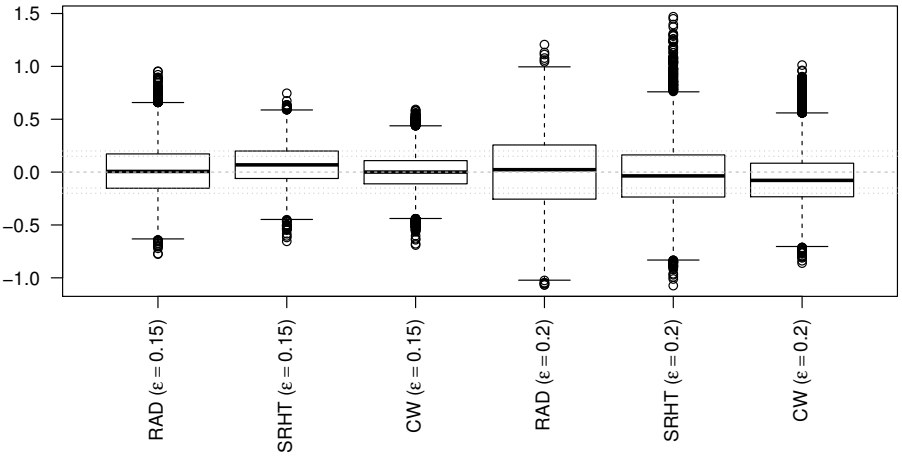

Figure 2.6: Difference of fitted values according to models based on the respective sketching methods and fitted values according to model based on the original bike sharing data set.

increases. The CW-sketch offers the lowest squared distances for both values of ε. The number of observations in the sketched bike sharing data set is higher when employing the CW method instead of the two others, this may be one reason for the favourable result.

Figure 2.6 shows the differences in the fitted values as boxplots. As in Section 2.4.5, the fitted values are based on the original data set X in both cases. We can see that the boxes are smallest for the CW-sketches compared to the other sketching methods when keeping ε constant. The majority of fitted values are close to 0, for both $\varepsilon = 0.15$ and $\varepsilon = 0.2$. Especially for $\varepsilon = 0.2$ and the RAD-sketch, some fitted values differ by an absolute value of around 1. The effect of ε becomes clear again: as ε grows, the target dimension of the sketched data set is smaller, but the approximation is not as well-fitting.

A more detailed analysis indicates that the highest deviations occur when the fitted number of bikes used is high. The relative difference between the fitted values is thus relatively low for the majority of observations.

As the next step, I compare the posterior distributions (as in Section 2.4.6). Figure 2.7 shows the boxplots of the MCMC-sample obtained for the two dummy variables that characterise the variable *weathersit*. Clear weather – which can be sunny or partly cloudy – serves as reference category. In the original data set, 66% of the observed hours show a weather situation that falls

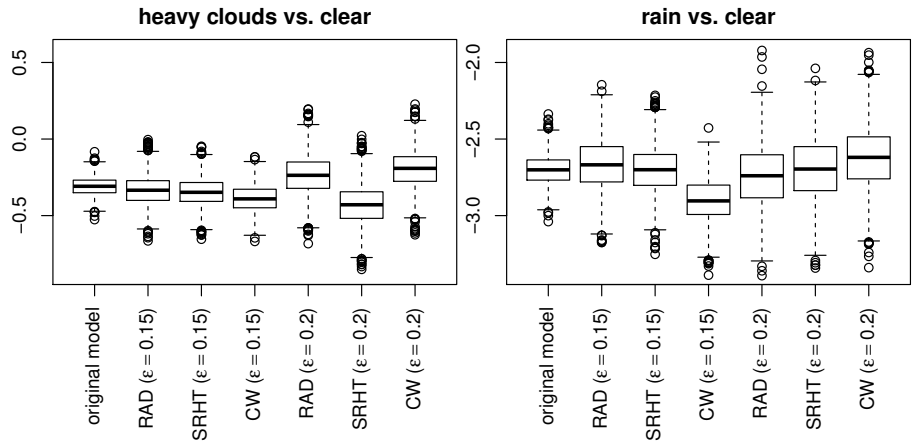

Figure 2.7: Boxplots of MCMC-sample for the two weather situation parameters based on the original data set and all sketches.

into category 1. 25% and 6% fall into the other two categories – cloudy weather and rain – respectively. On the left-hand side in Figure 2.7, we see the difference between clear and cloudy weather. There is a negative, but small effect on the number of rental bikes. For the model on the original data set as well as for all sketches with $\varepsilon = 0.15$, the 95% credible interval does not include 0. When changing ε to 0.2, this is only true for the SRHT-sketch. This illustrates that sketching introduces an area of uncertainty: when the credible interval of a variable is close to 0 in the model based on the original data set, the credible interval on a sketched data set may or may not include 0.

When the weather situation worsens and it is raining or there are thunderstorms present (right-hand side of Figure 2.7), the negative effect is considerably more pronounced. In both situations we can see that the results on the original data set and on the reduced data sets are close to one another. The approximation parameter ε plays an important role as higher values introduce more variation to the posterior distribution. The sketching method on the other hand does not seem to have a noticeable effect.

2.6 Conclusion

In the present chapter, my main focus lies on Bayesian linear regression. It is possible to apply random projections for frequentist linear regression without any problems. The three projection methods introduced in Section 2.3 constitute a good start for frequentist linear regression. Please note that there is extensive research on which random projections result in a good approximation in a useful running time, especially in Computer Science.

The sketch (2.4) on page 17 illustrates that our proposed method adds a step before the regression analysis takes place. In a way, the random projections serve as an elaborate pre-processing step that makes the analysis on data sets with a very large number of observations possible. This also means that they can be combined with a number of methods. In the current chapter, I only consider Bayesian linear regression models on the embedded data set, using Hamiltonian Monte Carlo techniques to approximate the posterior distribution. Note that this was done by choice, not by necessity. It is possible to employ different MCMC-techniques or other approximation methods. It may also be possible to combine random projections with other reduction techniques mentioned in Section 2.2. However, the benefits of two reductions are not immediately apparent.

In conclusion, the simulation study indicates that our proposed method works well for simulated data sets, which are generally well-suited for conducting Bayesian linear regression. But even with a high variance of the error term (and thus a relatively bad model fit), our proposal leads to results similar to those one would obtain on the original data set. The running time of the analysis with the proposed sketches is largely independent of n, giving advantages for very large n. Since the embeddings can be read in sequentially, it is not necessary to load the whole data set into the memory at once, which reduces the required memory.

Chapter 3

Generalisations to Other Regression Models

Verallgemeinert betrachtet sind Sneak Previews auch nur zufällige Projektionen.

(Unbekanntere Erkenntnisse des Projekts C4)

After considering linear regression in Chapter 2, I will now examine two generalisations. The first generalisation introduces an additional level of hierarchy to the model while the second replaces the normal distribution with a q-generalised normal distribution, also known as power exponential distribution. This chapter has been conceived in cooperation with Jonathan Rathjens and Steffen Müller, please also refer to their respective Master's theses (Müller, 2016; Rathjens, 2015).

3.1 Hierarchical models

3.1.1 Introduction

Hierarchical models introduce an additional level of hierarchy. This may be advantageous because information are present on different levels. Gelman et al. (2014, Chapter 15) mention scholastic achievement as an example, where covariate information about pupils may be on a family-level (e.g. parents' educational levels), on a class-level (e.g. influence by teachers), and on a school level (e.g. educational policies). For data arising from stratified or cluster sampling, hierarchical models offer a very natural choice.

Hierarchical models are related to the concept of exchangeability. Following Lindley and Smith (1972), exchangeability can and indeed should be assumed if the indices of the prior

distribution can be permuted without changing the prior itself, i.e., if the differences between variables are due to the observed data, not due to different roles prior to observing data.

In a way, exchangeability is similar to assuming independence between observations, but applies to prior information about variables, i.e. before we know anything about the data set, we assume identical prior information for every variable. This can be further illustrated using the hospital example Gelman et al. (2014, Chapter 5) present in their introduction to hierarchical models. In this example, the success (or survival) rates of different general hospitals in a certain operation are considered. Gelman et al. (2014) argue that it may be reasonable to assume hospitals are related to each other, leading to the introduction of a population distribution. We can further develop this example by assuming different groups of hospitals, for example general hospitals and hospitals with a specialisation related to the operation. The two different groups can reasonably be considered different prior to observing data, so it might be appropriate to include a hierarchical submodel for each of the groups.

In cases where exchangeability is present, Lindley and Smith (1972) argue that hierarchical models are the superior choice compared to standard frequentist linear regression. Whether we agree with this or not, hierarchical models offer important modelling options. Introducing one additional level of information, the so-called population distribution, is particularly useful for two reasons: the population distribution allows borrowing inferential strength from other observations and a shrinkage effect will occur.

Figure 3.1 illustrates the idea of a population distribution. The distributions in grey stand for individual distributions for every observation. They all differ, but stem from the same underlying distribution, given in black here.

In the following, I will only present models with one additional level of hierarchy, i.e. models with a prior distribution and a hyperprior. The extension to the more general case with to two or more levels of hierarchy follows in an analogous manner.

In Chapter 1, I introduced Model (1.10), which has a prior distribution $p(\underline{\beta})$ and a posterior distribution $p(\underline{\beta}|X, \underline{Y})$. In the hierarchical model a hyperparameter $\underline{\theta}$ is added, see Equation (3.1),

$$p(\underline{\beta}|X, \underline{Y}, \underline{\theta}) \propto p(\underline{Y}|X, \underline{\beta}, \underline{\theta})p(\underline{\beta}|\underline{\theta})p(\underline{\theta}). \tag{3.1}$$

Equation (3.1) shows the posterior distribution in a hierarchical model with one additional level. The posterior distribution is proportional to the likelihood multiplied with the prior distribution multiplied with the population distribution $p(\underline{\theta})$. On all levels, $\underline{\theta}$ is added as new

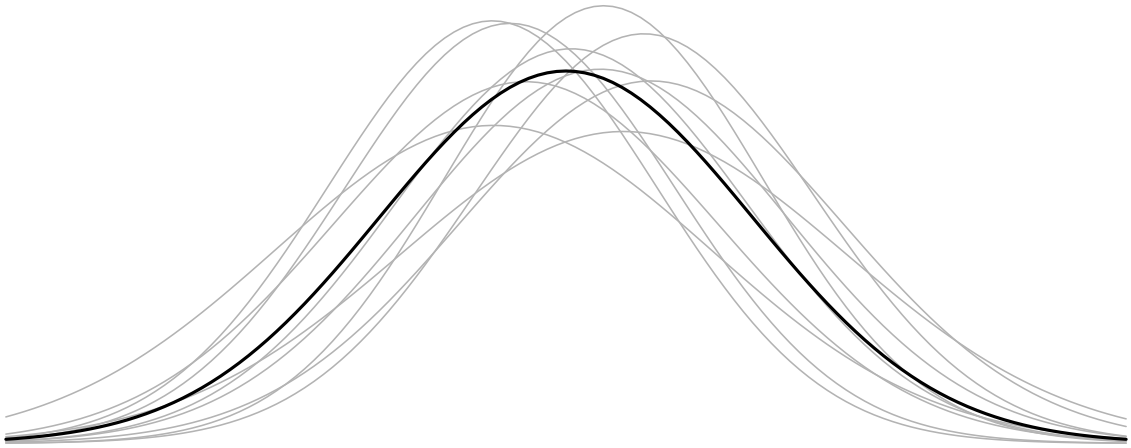

Figure 3.1: Schematic representation of a population distribution. The grey distributions are distributions for single observations. The black distribution is the underlying population distribution.

condition, while the population distribution $p(\underline{\theta})$ itself depends on hyperparameters which take over the role of the prior distribution in non-hierarchical regression models.

3.1.2 Sketching for hierarchical models

Besides an interesting introduction and motivation, Lindley and Smith (1972) also contains useful results concerning hierarchical linear models. The most interesting one in our context is a theorem that proves that the posterior distribution of $\underline{\beta}$ is independent of $\underline{\theta}$, but depends on the chosen hyperprior $p(\underline{\theta})$ of $\underline{\theta}$.

For this reason, I can employ the sketches introduced in Chapter 2 to hierarchical models without any changes. For the random projections, this means the hyperparameter is taken as part of a now more complex prior distribution. The theoretical guarantees introduced in Chapter 2 do not strictly cover the inclusion of a hyperprior in the model. However, as the posterior distribution of $\underline{\beta}$ only depends on the hyperprior, we can expect $\underline{\beta}$ to be well-recovered using the sketches from Chapter 2 as long as the hyperprior is chosen relatively non-informatively.

3.2 q-generalised regression

3.2.1 q-generalised normal distribution

Throughout Chapter 2, I employ normal distributions for both the prior distribution $p(\underline{\beta})$ and the likelihood $L(\underline{\beta}|X, \underline{Y})$. In this section, I replace the normal distributions with their q-generalised counterparts for the prior distribution (Section 3.2.2).

The normal distribution is based on the power of 2. Deviations from the expected value μ occur with relatively low probability, leading to light tails. The q-generalised normal distribution replaces the power of 2 with a power of $q, q \in (0, \infty)$. For $q > 2$, this leads to lighter tails compared to the normal distribution while $q < 2$ makes the tails heavier. The q-generalised normal distribution was first introduced by Subbotin (1923) and is formally introduced in Definition 4 following Nadarajah (2005).

Definition 4 (q-generalised normal distribution, following Nadarajah (2005))**.** *A random variable X with probability density function*

$$p(X) = \frac{q}{2\alpha\Gamma\left(\frac{1}{q}\right)} e^{-\frac{|X-\mu|^q}{\alpha^q}}, \tag{3.2}$$

where Γ is the gamma function $\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}dt$ for any positive $z \in \mathbb{R}^+$, follows a q-generalised normal distribution with location parameter $\mu \in \mathbb{R}$ and scale parameter $\alpha \in \mathbb{R}^+$, for $q \in (0, \infty)$.

X has an expected value of $\mathsf{E}(X) = \mu$ and a variance of $\mathsf{Var}(X) = \frac{\alpha^2\Gamma(3/q)}{\Gamma(1/q)}$. For the special cases of $q = 1$ and $q = 2$, X follows a Laplace distribution and a normal distribution, respectively. Figure 3.2 shows the q-generalised distribution for selected values of q. These distributions are scaled such that the expected value is 0 and the variance is 1 for all q.

Definition 4 and Figure 3.2 are concerned with the univariate case of the q-generalised normal distribution. Defining the multivariate case is straightforward.

3.2.2 q-generalised normal prior distribution

In this section, I generalise the prior distribution to include q-generalised normal distributions. The case of $q = 1$ is of special interest as it can be used to model a LASSO-regression model. The theoretical guarantees put forward in Chapter 2 in general do not apply any more as they are limited to arbitrary normal distributions and improper non-informative uniform distributions.

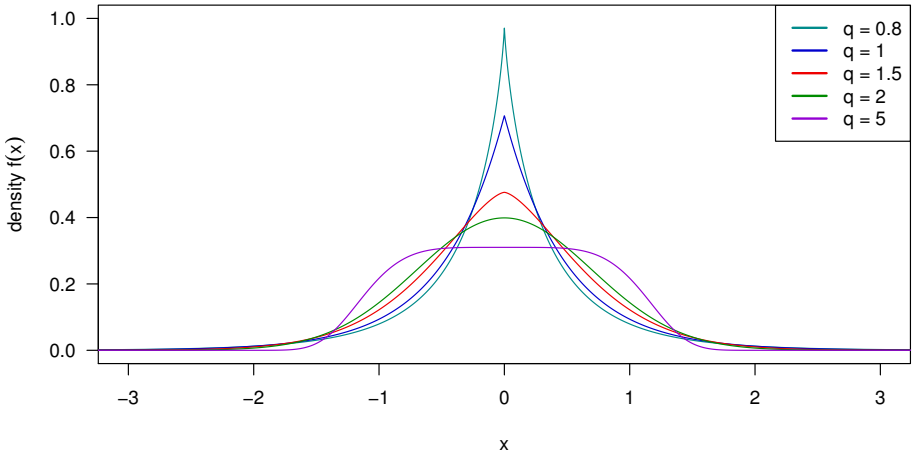

Figure 3.2: Probability density function of the q-generalised normal distribution for different values of q.

However, it is possible to employ the same sketching algorithms as before and we can expect the approximation to work well unless highly informative q-generalised normal distributions are used as prior distribution.

3.3 Simulation Study

3.3.1 Hierarchical models

The simulation study for the generalised cases is structured very similarly to Section 2.4. To generate the data sets, I start with drawing expected values V_j for every column $j = 1, \ldots, p$ of matrix X from a normal distribution with mean 0 and relatively high variance, $V_j \sim N(0, 25)$, $j = 1, \ldots, p$. In the next step, I draw realisations of the data matrix as $X_{ij} \sim N(V_j, 5^2)$, $i = 1, \ldots, n$, and in the last step, the realisations of $\underline{Y}$ are obtained via $Y_i \sim N(X_{i.}\underline{\beta}, \sigma^2)$. The parameters employed in the simulation study are the same as in Table 2.2. For the generalised regression models, I choose the parameters as $n \in \{10\,000, 100\,000\}$, $p \in \{10, 20\}$, and $\sigma^2 \in \{1, 25\}$. There is thus a total of eight data sets on which the simulation study is based.

To arrive at hierarchical regression or q-generalised regression, I only adjust the regression models to cover these changes, while the data sets remain structurally the same. To execute a hierarchical analysis, I use Model (1.10) as starting point, but now let $\underline{\beta}$ follow a normal distribution, i.e. $\beta_j \sim N(\xi, \rho^2), j = 1, \ldots, p$, where ξ and ρ^2 represent the population mean and the population variance, respectively. The prior distribution for ξ is a normal distribution, $\xi \sim N(2, 1)$ and the prior distribution for ρ^2 follows a Gamma distribution, $\rho^2 \sim \Gamma(4, 2)$. My

model thus contains one level of hierarchy where all components of $\underline{\beta}$ follow the same population distribution.

For the hierarchical models, I analyse both a non-informative and an informative prior distribution for the population distribution. The non-informative prior distribution consists of $p(\xi) \sim N(0, 100)$ and $p(\rho^2) \sim \Gamma(1, 0.01)$. The informative distribution is chosen to match the true population distribution used in the creation of the data sets, i.e. $p(\xi) \sim N(2, 1)$ and $p(\rho^2) \sim \Gamma(4, 2)$. In both cases, I employ a non-informative uniform distribution as prior distribution for the variance parameter σ^2.

The aim of employing different prior distributions is to analyse the difference between informative and non-informative priors on the original and the sketched data sets. However, adding (highly) informative prior information to the model may present a great difficulty when sketching the data set, especially if the prior information contradicts the information in the likelihood. For that reason, I opt for the case of the prior information representing the true underlying distribution even though it is unlikely in practice.

For q-generalised regression I can employ Model (1.10) and change either the prior distribution to a q-generalised normal distribution. I choose values of $q \in \{0.75, 1, 1.25, 1.5, 1.75\}$. This is due to earlier analyses in Müller (2016), which indicate that values of $q > 2$ behave similarly to the special case of linear regression with $q = 2$. For all these cases, I use a normal distribution as likelihood.

To obtain sketches of the data sets, I employ all three sketching methods introduced in Section 2.3 with $\varepsilon = 0.1$ and $\varepsilon = 0.2$, respectively, for the hierarchical models as well as the models with a q-generalised normal prior distribution. Table 3.1 contains the resulting numbers of observations k for the sketched data sets.

setting	RAD	SRHT	CW
p=10, $\varepsilon = 0.1$	660	660	256
p=10, $\varepsilon = 0.2$	2638	2638	1024
p=20, $\varepsilon = 0.1$	4761	4761	4096
p=20, $\varepsilon = 0.2$	1599	1599	1024

Table 3.1: Number of observations k of the sketches for different values of variables p and approximation parameter ε.

I now compare the posterior distributions of the parameters of interest resulting from the original data sets and the sketched data sets, respectively. Figure 3.3 contains exemplary boxplots of the posterior distribution of parameter β_2 for an original data set with $n = 100\,000, p = 10$, and

$\sigma = 5$. The left-hand side depicts the posterior distributions for the models with non-informative prior distribution, while the right-hand side depicts the results for models with informative prior distribution. Figure 3.3 indicates that the choice of prior distribution has little influence on the posterior distribution. As in Chapter 2, there may be some differences in location between posterior distributions for original data sets and sketched data sets, but there seem to be no systematic biases. Again, the sketching methods introduce some extra variation which seems to grow as ε grows.

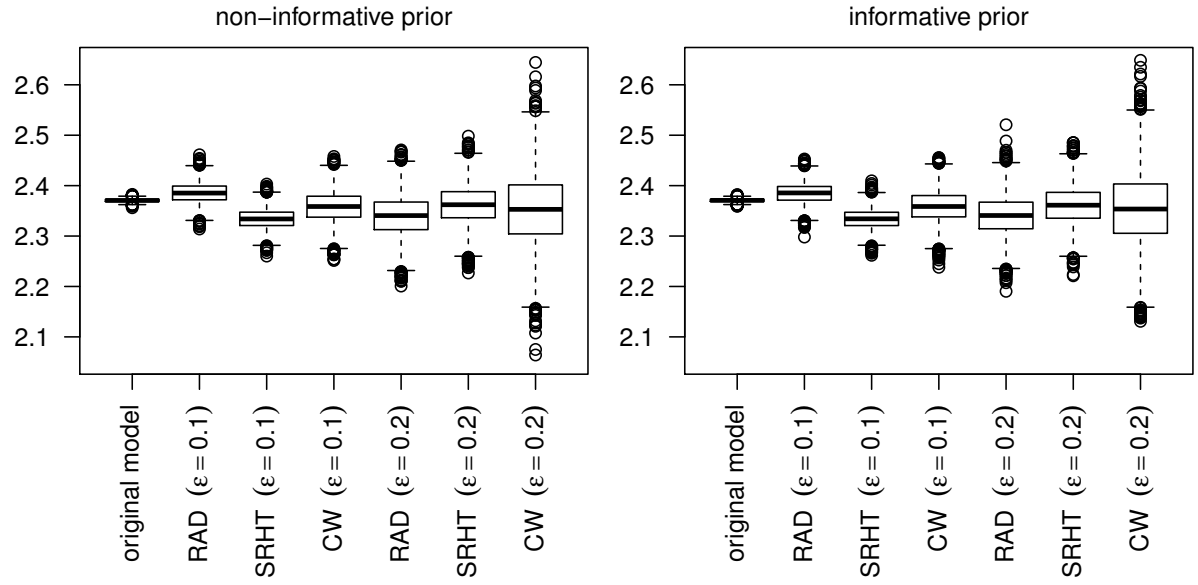

Figure 3.3: Boxplots of MCMC-sample for the parameter β_2 of a hierarchical model with non-informative prior (subfigure (a)) and informative prior (subfigure (b)) based on the simulated data set with $n = 100\,000$, $p = 10$, $\xi = 5$ and respective sketches.

Figure 3.4 shows the corresponding boxplots for the hyperparameter μ for models with both non-informative and informative prior distribution. The dispersion of the hyperparameter is a lot higher compared to the dispersion of the regular parameters. However, sketching does not seem to influence the variation in the distribution of the hyperparameter, all posterior distributions based on sketches seem to be virtually identical to the posterior distribution on the original data set. There is a difference between the two models, however: The higher variation present in the non-informative prior distribution for μ compared to the informative prior distribution can also be found in the two respective posterior distributions.

Similarly, Figure 3.5 contains boxplots for the hyperparameter ξ, which represents the population mean, for two data sets with different values of σ. The values of n and p are identical for the two data sets and both models employ a non-informative prior. The hyperparameter ξ

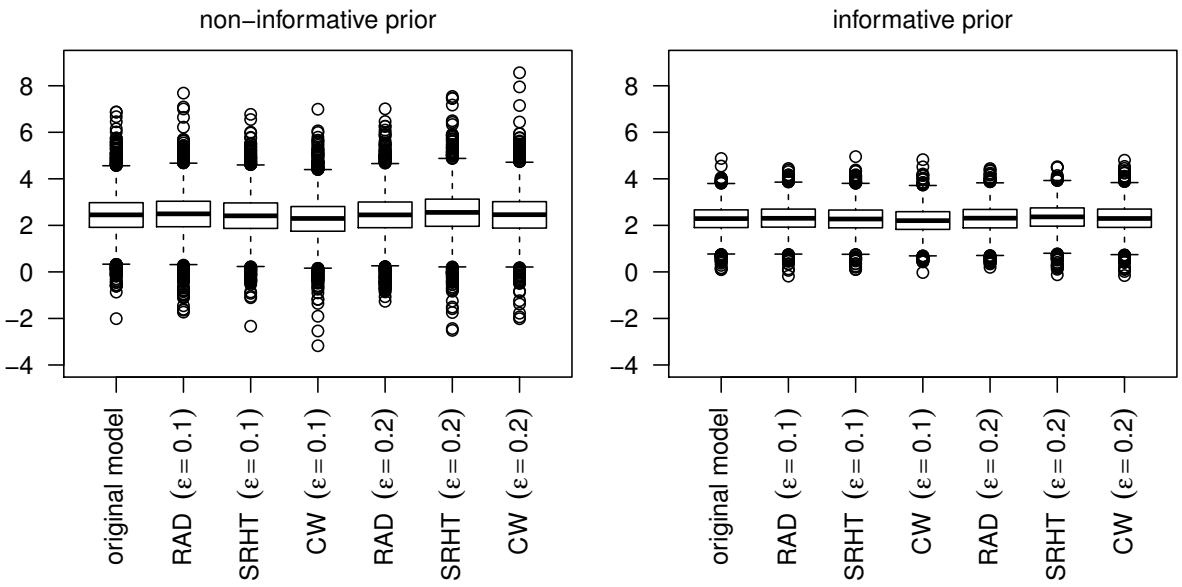

Figure 3.4: Boxplots of MCMC-sample for the hyperparameter μ of a hierarchical model with non-informative prior (subfigure (a)) and informative prior (subfigure (b)) based on the data $n = 100\,000$, $p = 10$, $\xi = 5$ and respective sketches.

is not well-recovered by the random projection. The location of the posterior distributions of ξ on the sketched data sets are inflated by a factor of approximately $\sqrt{\frac{n}{k}}$ while the dispersion also increases, which indicates a multiplicative effect. Figure 3.5 indicates that the posterior distributions are similar for the sketching methods RAD and SRHT, but differ for CW-sketches. This is probably due to the difference in the target dimension k between RAD and SRHT on the one hand and CW on the other hand. The effect of an increase in σ, which I set to a value of 1 in subfigure (a) of Figure 3.5 and 5 in subfigure (b), seems to affect ξ directly in a multiplicative way. The values in subfigure (b) are approximately five times higher than in subfigure (a).

Figure 3.6 shows the difference between fitted values based on the original data set and fitted values based on the respective sketches. Similarly to Chapter 2, most of the differences are close to 0, which indicates that conclusions drawn about predicted values of $\underline{Y}$ are well-recovered by the sketching methods. This is a reflection of the well-recovered posterior distribution of $\underline{\beta}$. The population mean ξ, which is not well-recovered, does not directly influence the predicted values $\underline{\hat{\beta}}$.

To recapitulate, the simulation study on hierarchical models indicates that the models on the sketched data sets recover the posterior distribution on the first level as well as in the simulation study on non-hierarchical Bayesian models in Chapter 2. As seen before, the method introduces some additional variation.

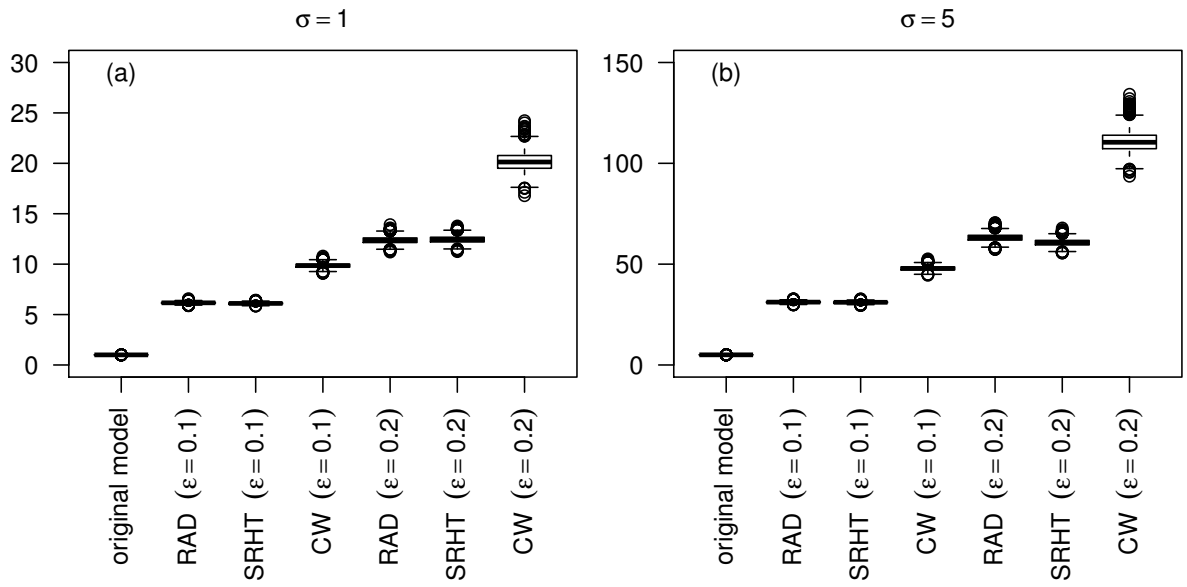

Figure 3.5: Boxplots of MCMC-sample for the hyperparameter ξ of a hierarchical model with non-informative prior for two different data sets with $n = 100\,000$, $p = 10$, $\sigma = 1$ (subfigure (a)) and $\sigma = 5$ (subfigure (b)) and the respective sketches.

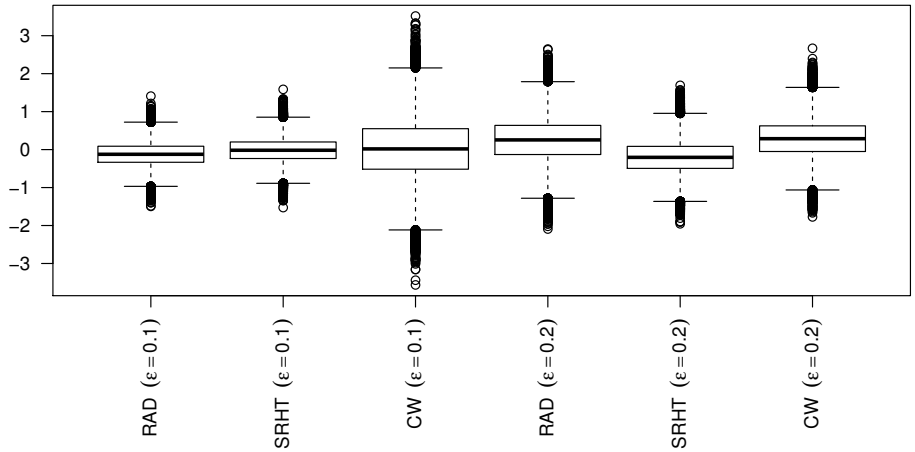

Figure 3.6: Difference of fitted values according to hierarchical models based on the respective sketching methods and fitted values according to hierarchical model based on the original data set with $n = 100\,000$, $p = 10$, $\xi = 5$.

3.3.2 q-generalised regression models

I now consider a non-hierarchical model that employs as prior distribution a q-generalised normal distribution instead of the normal distribution used before. In total, I consider six different values of q, $q \in \{0.75, 1, 1.25, 1.5, 1.75, 2\}$. The special cases $q = 2$ and $q = 1$ stand for the normal distribution and the Laplace distribution or double exponential distribution, respectively. Analogously to the hierarchical case, I consider both a model with non-informative prior and

a model with highly informative prior distribution. Here, the difference lies in the variance of the prior distribution. To model the non-informative prior, I employ a q-generalised normal distribution with parameters $\mu = 2$ and $\alpha = 100$. For the highly informative prior distribution, $\mu = 2$ stays the same, but the scale parameter α is set to $\alpha = 0.1$. This may seem an extreme choice, but in a setting with very large data sets, mildly informative prior distributions tend to not have a lot of influence on the posterior distribution due to the large number of observations the likelihood is based upon.

In both cases, the prior distribution draws the posterior distribution towards 2, which is not necessarily supported by the likelihood. This is a conscious choice. The aim is to examine cases where the prior distribution is informative, but contains information that contradict the likelihood. It is of interest how this set-up influences the results especially on sketched data sets.

Figure 3.7 depicts the results for parameter β_2 for different values of q based on the original data set and sketched data sets where sketches are based on the CW-sketch with a value of $\varepsilon = 0.1$. The result for the original data set (left-hand side of Figure 3.7) shows virtually identical posterior distributions for all values of q. For all six prior distributions, the posterior distribution's median lies around -0.07. The 2.5%- and 97.5%-quantiles are around -0.09 on the lower end and -0.05 on the upper end. According to the original model, β_2 can be considered an influential variable in the regression model, albeit the posterior distribution lies close to 0. On the right-hand side of Figure 3.7, the respective posterior distributions based on the sketched data sets are shown (CW-sketch with $\varepsilon = 0.1$). Again, the posterior distributions look almost identical regardless of the values of q. Based on the sketches, however, we would not consider β_2 an influential variable. All posterior medians are around -0.014, the 2.5%- and 97.5%-quantiles take values of -0.08 and 0.05, respectively. The 95% credible intervals thus contain 0. The models with non-informative prior distributions present a case where a variable is close to, but different from 0 according to the 95% credible interval on the original data set, but according to analysis on the sketched data set may well be simply vary around 0. Different values of q have no influence on the posterior distribution in this setting.

I now compare these results to a case with highly informative prior distributions. For these models, Figure 3.8 shows the posterior distributions for β_2 based on models on the original data set (left-hand side) and based on models on the sketched data sets (right-hand side), both for different values of q. On the original data set, the posterior distribution of β_2 lies below 0 for all values of q. There is a small, but distinct effect of q: as the values of q grow, the posterior distribution is drawn further towards the prior distribution. In this concrete example, this means

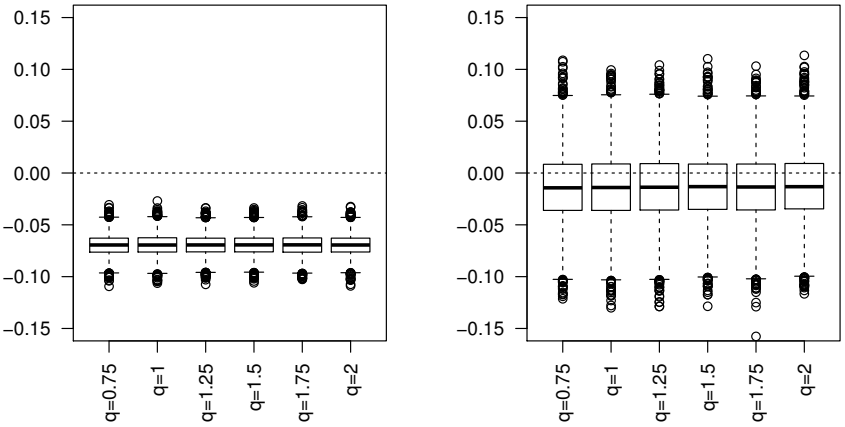

Figure 3.7: Boxplots of the MCMC-sample for parameter β_2 with different non-informative q-generalised normal distributed prior and normal distributed Likelihood on the original data (left-hand side) and sketched data (right-hand side). Sketches are based on the CW-sketch with $\varepsilon = 0.1$.

that the posterior distribution is closer to 0 as q grows. This effect is even more obvious for the analyses on the sketched data set. For values of $q \leq 1.5$, the posterior distribution includes 0, indicating an unclear role of the variable for the importance of the model. For $q = 1.75$ and $q = 2$, the 95% credible intervals of posterior distribution do not include 0, we would therefore conclude that the dependent variable grows as β_2 increases.

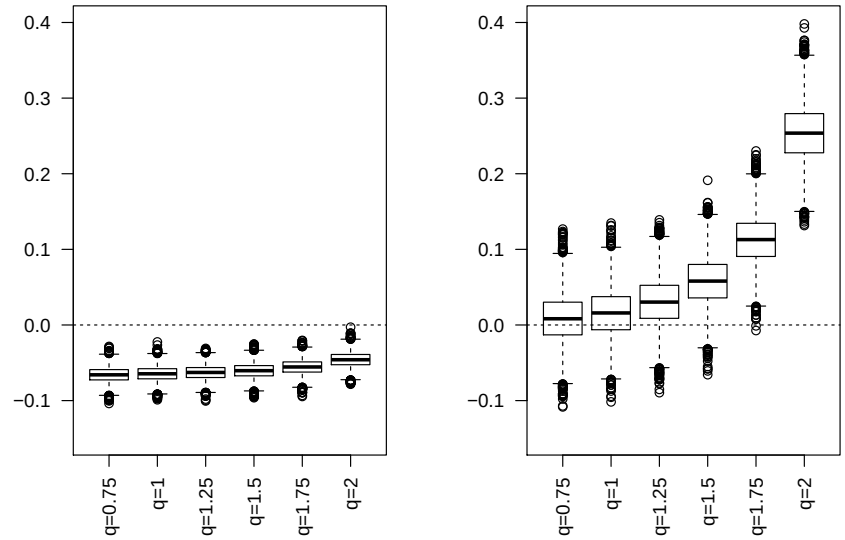

Figure 3.8: Boxplots of the MCMC-sample for parameter β_2 with different informative q-generalised normal distributed prior and normal distributed Likelihood on the original data (left-hand side) and sketched data (right-hand side). Sketches are based on the CW-sketch with $\varepsilon = 0.1$.

In conclusion, the empirical results suggest that random projections are suitable for regression models that employ the q-generalised normal distribution as prior distribution. For non-informative prior distributions, the behaviour observed is very similar to the results found in Chapter 2. Specifically, regression analyses on sketched data sets introduce some additional variation into the posterior distributions. This may lead to extreme cases where a variable would be considered important in the original model in the sense that the 95% credible intervals do not contain 0, but the credible interval for the same variable on the sketched data set does contain 0 (compare Figure 3.7). Apart from this caveat, random projections are able to recover the posterior distribution well and the value of q in the interval $q \in [0.75, 2]$ does not seem to have any influence on the result.

For highly informative prior distributions, the choice of q has a distinct effect on the posterior distribution. This effect is visible for models both on the original data sets and on sketched data sets. As q increases, the posterior distribution is drawn further towards the prior distribution. In other words, small values of q around 1 make the posterior distribution less reactive to prior information that contradict the information given in the likelihood. As mentioned in Section 3.2.1, employing a q-generalised normal distribution as prior distribution changes the heaviness of the tails. This is because the distance of values from the centre of the distribution is exponentiated to the power of q, which makes values that are far from the centre increasingly unlikely for increasing values of q. In the simulation study, this leads to a posterior distribution that is more heavily influenced by the prior distribution as the differences between likelihood and prior distribution are given increasingly more relative importance as q grows.

3.4 Bicycle rental data set

For the hierarchical analysis of the bicycle rental data set introduced in Section 2.5, I now introduce a model that reflects the hierarchical structure. To that end, variables that logically form a unit are given a common hyperprior. In detail, this means that all dummy variables that are employed to model the factor variables are considered as exchangeable and thus given the same hyperprior distribution. The three quantitative variables *atemp*, *hum*, and *windspeed* are also taken to be exchangeable and assigned one common hyperprior.

As before, I employ the square root of the number of rented bikes $\sqrt{Y_i}, i = 1, \ldots, n$, as target variable. In the model, I assume a normal distribution with expected value $\underline{\beta}$ and standard deviation of ξ for $\underline{Y}$. As prior distributions for $\beta_j, j = 1, \ldots, p$, the components of $\underline{\beta}$, I employ normal

distributions with expected values $\mu_h, h = 1, \ldots, 7$, and standard deviation $\sigma \sim \Gamma(1, 0.01)$, where h (or $h(j)$) is the appropriate hierarchical effect for variable j. Each of the μ_h's has a normal distribution with expected value 0 and standard deviation 100 as prior distribution.

Figure 3.9 on page 52 shows the results of the hierarchical model on the bike rental data set. The posterior distributions for the effects on the lowest level are similar to the results of the non-hierarchical model. All of the effects on the lowest level show small variances and high mass concentration around the median. The effects on the upper level, on the other hand, seem to represent the locations of the population densities well and exhibit high variances around their location. The variance seems to depend on the number of (dummy) variables that the hyperdistribution represents: the variance is highest for hyperdistributions of *holiday* and the intercept, both of which are only based on one effect on the lower level. On the other hand, the variance of the hyperdistribution is relatively low the the variable *hour*, which consists of a total of 23 dummy variables.

Figure 3.10 exemplarily shows the boxplots of the posterior distributions of all variables related to *weekday* for both the original data set and the sketched data set. The variables include six dummy variables on the lower level and one hypervariable on the upper level. The posterior distribution of the hyperparameter only slightly differs between the sketched data set and original data set. For the dummy variables on the lower level, we can again see that the location is similar with some slight changes, but sketching introduces some additional variation.

3.5 Conclusion

The current chapter describes how the sketching techniques introduced in Chapter 2 can be applied to more complicated models, in particular models where the prior distribution contains hierarchical information and models where the prior distribution employs a q-generalised normal distribution. The model class of q-generalised normal distributions

In contrast to Chapter 2, this chapter is mainly based on empirical results. However, some theoretical guarantees can be carried over to the models with more complicated prior distribution considered here. For some hierarchical models, the posterior distribution on the lower level is independent of the posterior distribution on the higher level, but depends on the prior distribution of the hyperparameter. While this does not give theoretical guarantees, it does allow the application of random projections on a broad class of hierarchical models. Similarly, when changing the prior distribution of the model, some of the guarantees for linear regression can

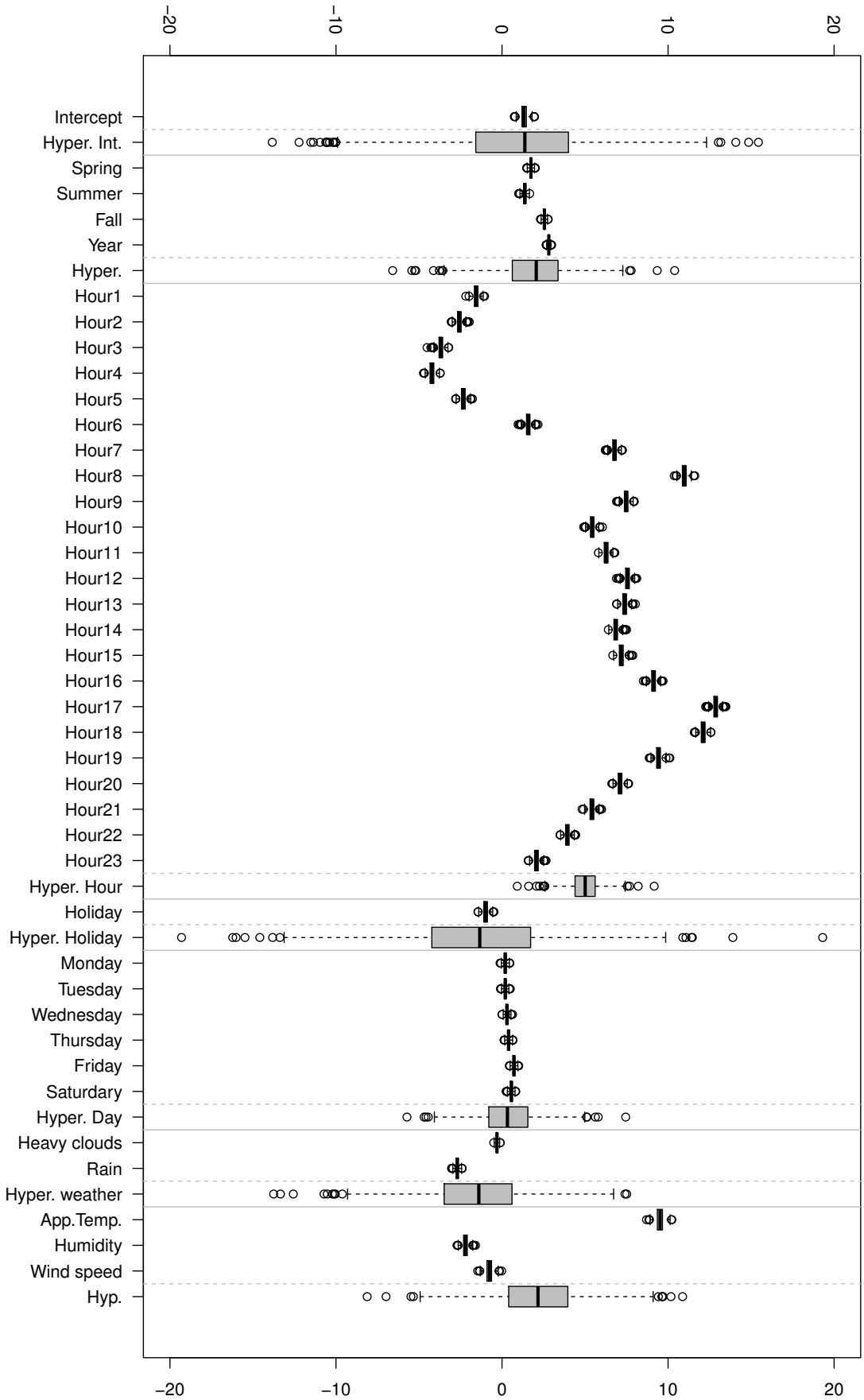

Figure 3.9: Boxplots of the MCMC-samples for all β parameters and the hyperparameters of the bike sharing data set based on the original hierarchical model.

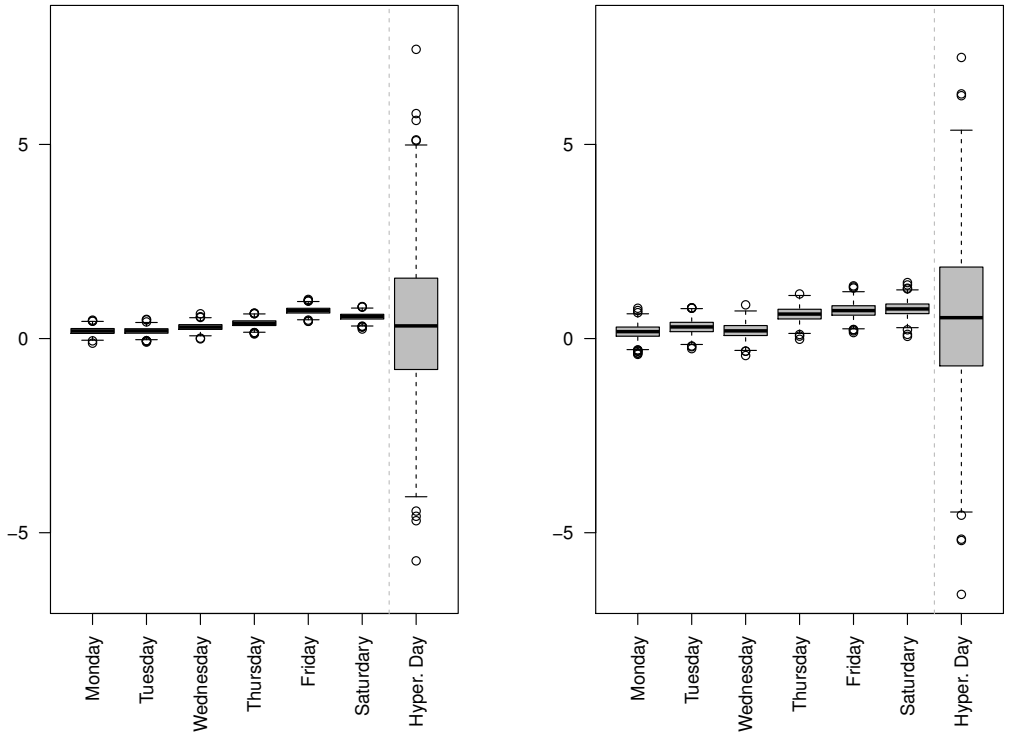

Figure 3.10: Boxplots for the posterior distributions of the variables related to *weekday* for the hierarchical model based on the original data set (left-hand side) and the sketched data set (right-hand side).

be carried forward. In both cases, the guarantees may potentially not hold, but it is hard to formally describe when this is the case. Informally speaking, the random projections can generally be expected to work well unless prior distributions are highly informative and contradict the information in the data. In such cases, randomly projecting the data set into a subspace may lead to a large approximation error. However, from a modelling perspective, such a situation should only arise in very rare circumstances.

Changing the likelihood from normal distributions to q-generalised normal distributions is also possible. However, the sketching methods introduced in Chapter 2 are too coarse in this case. In Müller (2016), the CW sketching method is employed as the first step of a two-step procedure. This first step serves to form a well-conditioned basis, which is then sampled from in the second step. Models with q-generalised normal distribution as likelihood are not considered in this thesis, but remain a topic in the research project C4 of SFB876 and will be covered in a forthcoming manuscript.

Chapter 4

The Merge & Reduce-Technique for Regression Models

When did my friends slip right through my fingers and you, you were all I ever knew?
You were all I ever knew – laughing in the afterglow.

(The Crookes – Afterglow)

The Merge & Reduce-technique presents a different option for handling very large data sets. While in Chapters 2 and 3, the focus lies on a reduction of the number of observations and a fast subsequent analysis, in this chapter, it lies on obtaining a stable result for very large data sets and also in streaming settings. I will first introduce the background of Merge & Reduce and its use in Computer Science as well as related work (Section 4.1) before presenting our proposal on how to transfer Merge & Reduce to application on statistical models (Section 4.2). After that, I will put forward three different approaches to Merge & Reduce in a regression setting (Section 4.3). This is followed by a short introduction of generalised linear models (GLMs) in Section 4.4. The three approaches are empirically evaluated on simulated data sets in Section 4.5 and on the bicycle usage data set in Section 4.6. Section 4.7 finishes the chapter with remarks on the results and what they imply regarding the practical usability of Merge & Reduce.

4.1 Background and related work

In Computer Science, Merge & Reduce is mainly employed as a way helping to obtain summaries of large data sets. It was first introduced by Bentley and Saxe (1980) as a general method for extending static data structures to handle dynamic insertions. Later on, the focus has been on

the data streaming setting where Merge & Reduce can be employed to work on coresets, small subsamples that represent the full data set well with regard to certain criteria. Examples of these adaptations can be found in Agarwal et al. (2004); Har-Peled and Mazumdar (2004).

Coresets have been studied extensively in the Computer Science literature as they present a scalable tool for data aggregation and reduction for many problems, including many statistical problems. Recently, Merge & Reduce has become a standard technique in the coreset literature and has been used to design efficient algorithms for the analysis of very large data sets, both in streaming settings and for distributed environments. Some examples are coresets for ℓ_1-regression (Clarkson, 2005; Clarkson et al., 2016; Sohler and Woodruff, 2011), ℓ_2-regression (Cohen et al., 2015b; Drineas et al., 2006, 2008; Li et al., 2013), ℓ_p-regression (Dasgupta et al., 2009; Woodruff and Zhang, 2013), M-estimators (Clarkson and Woodruff, 2015a,b), and generalised linear models (Huggins et al., 2016; Molina et al., 2018; Munteanu et al., 2018; Reddi et al., 2015; Tolochinsky and Feldman, 2018). Phillips (2017) offers a recent and extensive survey, while Munteanu and Schwiegelshohn (2018) introduce coresets and related methods on a technical level. Merging and reducing techniques similar to Merge & Reduce were also employed in the area of physical design for relational databases (Bruno and Chaudhuri, 2007).

Our aim is to develop a Merge & Reduce-technique that can be employed directly on statistical models, with the concept of Merge & Reduce on data sets or coresets serving as a basis. To that end, we aim at building statistical models on subsets of the data and need to find ways of merging and reducing these models. We will focus on regression models here, but the idea is transferable to other statistical models. Our idea is to iteratively load as many observations into the memory as is feasible or desirable. On each of these subsets, a regression model is calculated. Models are merged and their complexity reducing according to certain rules that are introduced in Section 4.3, eventually resulting in a final model that combines the information from all subsets. Merge & Reduce leads to stable results where every observation enters the final model with equal weight, thus ensuring that the outcome is invariant to permutations of the blocks. When employing the technique for statistical models that are not regression models, the merging and reducing strategies need to be changed accordingly. Finding suitable strategies is not trivial.

A recent idea from the Statistics literature is similar in spirit. Law and Wilkinson (2018) studied composable models for Bayesian analysis of streaming data. Their focus lies on highly heterogeneous streams from sensor networks, which they model using partially observed Markov processes. The authors propose composition procedures of those models similar to the merging procedures that we develop here in the Merge & Reduce framework. However, due to their focus

on heterogeneous streams, they address the asynchrony of sampling frequencies in practical streaming and distributed settings and thus have a different scope.

4.2 The Merge & Reduce-Principle

Merge & Reduce consists of four steps that are carried out repeatedly. Here, I present the four steps of our novel approach of employing Merge & Reduce on statistical models. The four steps are:

1. reading a block of data,
2. performing a statistical analysis on the current data,
3. storing a model that summarises the analysis,
4. merging models following a tree structure (see Figure 4.1 on page 57) while ensuring their complexity does not increase.

Before carrying out Merge & Reduce, it is necessary to set the number of observations per block to a number n_b. In a streaming or Big Data setting, the total number of observations n may typically be unknown beforehand. For that reason, n_b does not depend on n, but mainly on memory requirements or similar considerations. The total number of blocks B is then also unknown beforehand, however it is known that $n \leq n_b B$.

How to summarise the statistical analysis and how to merge models are not trivial questions. The answers heavily depend on the statistical method employed. To the best of our knowledge, Merge & Reduce has not been utilised in a regression context yet. We will propose merge-steps and reduce-steps for linear regression as well as for generalised linear models with an application to Poisson regression.

In our implementation, every model summarises the respective regression analysis. Additionally, all models contain meta-information, specifically the level of the model l and the number of observations the model is based upon, $n_{b,i}$, where i depends on l. For the sake of brevity, the index is indicated as i and not as i_l. Every model that directly summarises a statistical analysis has a level l of 1. For all blocks $b = 1, \ldots, B-1$, the number of observations is n_b. For the last block B, the number may be smaller. Merging two models that are not based on the same number of observations does not pose a problem provided appropriate weighting is employed.

Figure 4.1 illustrates the principle of Merge & Reduce. Data is read blockwise until the desired number of observations or the end of the file is reached and a model is built on that block

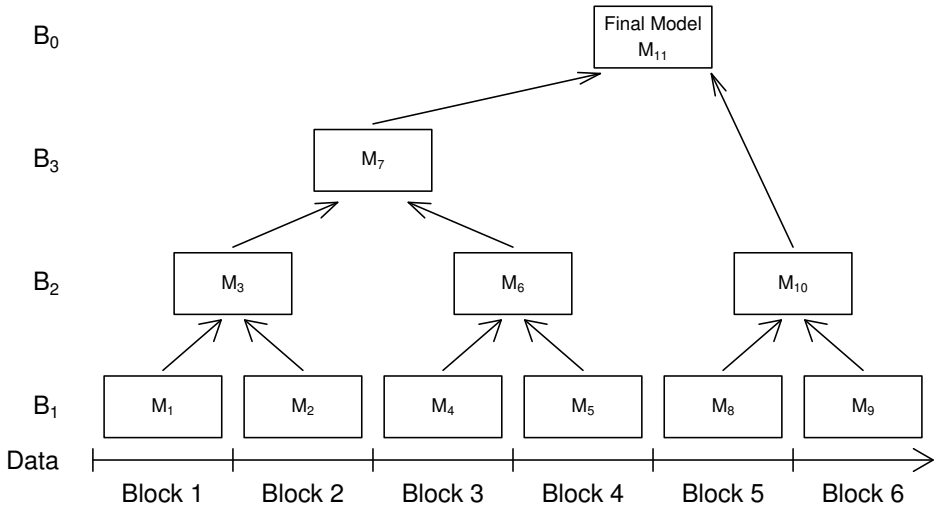

Figure 4.1: Illustration of the principle of Merge & Reduce for statistical models.

of data. In a setting with a very large number of observations n or in a streaming setting, the data is deleted as soon as the model has been built. Models are deleted as soon as their information are passed on to the next level, i.e. afer merging them with another model. This keeps the memory required for the storage of observations constant, while the memory required for the storage of models only logarithmically depends the number of blocks that have to be analysed, $O\left(\ln\left(\frac{n}{n_b}\right)\right)$. As the parameter n_b is chosen independently of n and represents a constant once chosen, this leads to a number of models that logarithmically depends on n, $O(\ln(n))$. To underline this logarithmic dependency, storage models B_1, B_2, B_3 and worker model B_0 are included in Figure 4.1. They will be formally introduced in Algorithm 2.

To achieve such a weak dependency on n, models are merged as soon as two models with the same level become available. For the first time, this happens when M_2 is stored, M_1 and M_2 can then immediately be merged to form M_3. M_4 is not merged with M_3 as both models are on different levels. However, when M_5 is calculated, it is merged with M_4 into M_6, which then in turn is merged with M_3 to form M_7. After merging two models, the new merged model has to be reduced in such a way that the complexity of the models does not increase with the number of merge-operations. Constant complexity of the models on every level is an important part of the technique of Merge & Reduce as indicated by the name. There are different reasons that justify this importance. One is the ability to bound the memory requirements. If the model complexity increased with every merge-step, the memory required would increase much quicker as n grows. The number of merge-steps and thus the complexity of the final model would be

Algorithm 1: Concept of Merge & Reduce for statistical models

$i \leftarrow 0$
level $l \leftarrow 1$
choose desired number of observations per block, n_b
repeat
  $i \leftarrow i + 1$
  read next $n_{b,i}$ observations // $n_{b,i} = \min(n_b, \texttt{\#observations remaining in file})$
  perform regression analysis on current observations
  $S_i \leftarrow$ summary statistics
  create new Model $M(l, i, n_{b,i}, S_i)$
    // level l, index i, number of observations $n_{b,i}$, summary statistics S_i
  delete current observations
  $j \leftarrow i$
  $k \leftarrow l$
  while $M(k, j-1, n_{b,j-1}, S_{j-1})$ *exists* **do**
    $M(k+1, \frac{j}{2}, (n_{b,j-1} + n_{b,j}), S_{\text{m\&r}}) \leftarrow$ **merge**
     $(M(k, j-1, n_{b,j-1}, S_{j-1}), M(k, j, n_{b,j}, S_j))$
                                                  // merge-step includes reduce-step
    delete $M(k, j-1, n_{b,j-1}, S_{j-1})$
    delete $M(k, j, n_{b,j}, S_j)$
    $j \leftarrow \frac{j}{2}$
    $k \leftarrow k + 1$
until $n_{b,i} < n_b$
while $j > 1$ **do**
  if j *even* **then**
    $M(k+1, \frac{j}{2}, (n_{b,j-1} + n_{b,j}), S_{\text{m\&r}}) \leftarrow$ **merge**
     $(M(k, j-1, n_{b,j-1}, S_{j-1}), M(k, j, n_{b,j}, S_j))$
                                                  // merge-step includes reduce-step
    delete $M(k, j-1, n_{b,j-1}, S_{j-1})$
    delete $M(k, j, n_{b,j}, S_j)$
    $j \leftarrow \frac{j}{2}$
    $k \leftarrow k + 1$
  else
    $M(k+1, \frac{j+1}{2}, n_{b,j}, S_j) \leftarrow M(k, j, n_{b,j}, S_j)$
    delete $M(k, j, n_{b,j}, S_j)$
    $j \leftarrow \frac{j+1}{2}$
    $k \leftarrow k + 1$
return *Final Model*

of order $O\left(\frac{n}{n_b}\right)$, which is infeasible for large data sets. It would also introduce a dependency between the complexity of the final model and the number of observations per block n_b, which is not a desirable property.

Algorithm 1 shows the Merge & Reduce algorithm in pseudo-code. Figure 4.1 might lead to the assumption that the number of observations is a multiple of the observations per block,

$n = 8n_b$. When $n \neq 2^m n_b, m \in \mathbb{N}$, which is usually the case, it is necessary to subsequently merge all remaining models in order to obtain the final model (the final while-loop). In such cases, the last model on each level is based on fewer observations than the other models. Appropriate weighting thus becomes important.

There is an alternative way of describing the algorithm that focuses more on the efficient memory usage. Initially, there are $c = \lceil \log_2(n/n_b) \rceil$ empty models $M_1, \ldots, M_c$ and an empty worker model M_w. The worker model stores information about the current batch of data while the c models stand for one level of information each. After reading and analysing a new batch of data, the summary statistics are stored in the worker model. If model M_1 is currently empty, the information in M_w are stored there. Otherwise, M_1 and M_w are merged to become the new worker model while M_1 is deleted. This continues up the levels until the algorithm encounters an empty model. After analysing the last batch of data, one additional iteration of all levels is necessary to obtain the final model M_c. If a model has to be merged with an empty model, the level i will be increased without a change of the summary statistics. Such a case is depicted in Figure 4.1, where Model 10 is first merged with an empty model before it is merged with Model 7 to obtain the final model. As in the first description, weighting according to the number of observations ensures that all observations hold the same importance for M_c. Algorithm 2 gives the second description in pseudo-code.

Algorithms 1 and 2 describe the same algorithm, the first version places more focus on merging only models that represent the same level of information while the second version places emphasis on the small number of models that need to be stored in memory.

Care is required when applying Merge & Reduce to a data set with factor variables as independent variables or as dependent variable. By dividing the data set into blocks, some values of the factor variables may occur very rarely or not at all in some blocks. This can lead to imbalanced models that are hard to estimate or even to undefined estimates for some values of the factor. This makes it difficult to employ Merge & Reduce for models with factors unless additional assumptions are made. I will go into more details about related modelling problems in Section 4.6.

4.3 Merging approaches

In this section, I will introduce three different approaches that can be utilised for Merge & Reduce. Each of the approaches consists of a merge-step and a reduce-step and is appropriate

Algorithm 2: Concept of Merge & Reduce for statistical models as a binary counter

choose desired number of observations per block, n_b
$c \leftarrow \lceil \log_2(n/n_b) \rceil + 1$
create empty models $M_1(0, \texttt{NA}), \ldots, M_c(0, \texttt{NA})$ `// storage models`
create empty model $M_w(0, \texttt{NA})$ `// worker model`
repeat
  read next $n_{b,w}$ observations
  `//` $n_{b,w} = \min(n_b, \texttt{\#observations remaining in file})$
  perform regression analysis on current observations
  $S_w \leftarrow$ summary statistics `// choice of user`
  $M_w \leftarrow M(n_{b,w}, S_w)$ `// number of observations` $n_{b,w}$`, summary statistics` S_w
  delete current observations
  $i \leftarrow 1$
  while $M_i(n_{b,i}, S_i)$ *exists* **do**
    $M_w(n^*_{b,w}, S^*_w) \leftarrow \textbf{merge}(M_w(n_{b,w}, S_w), M_i(n_{b,i}, S_i))$ `// merged worker model`
    delete $M_i(n_{b,i}, S_i)$
    $i \leftarrow i + 1$
  $M_i(n_{b,i}, S_i) \leftarrow M_w(n_{b,w}, S_w)$
  delete $M_w(n_{b,w}, S_w)$
until $n_{b,w} < n_b$
for $i \in \{2, \ldots, c\}$ **do**
  $M_i(n_{b,i}, S_i) \leftarrow \textbf{merge}(M_i(n_{b,i}, S_i), M_{i-1}(n_{b,i-1}, S_{i-1}))$
  delete $M_{i-1}(n_{b,i-1}, S_{i-1})$
return *Final Model* M_c

or advantageous in different situations. From here on, I will call them M&R approach 1, 2, and 3, respectively. All approaches will be empirically evaluated in Sections 4.5 and 4.6. M&R approaches 1 and 2 are general approaches that can handle broad classes of regression models. They are meant for frequentist and Bayesian analyses, respectively. M&R approach 3 is suitable when frequentist linear regression is conducted and recovers the full model very well and efficiently in such cases.

4.3.1 Estimate and standard error as summary values (M&R approach 1)

In a frequentist regression setting, linear models as well as generalised linear models are typically characterised employing two components: an estimate $\underline{\hat{\beta}}$ and its standard error $\text{se}_{\underline{\hat{\beta}}}$. It is also essential for prediction as the estimate $\hat{\beta}_j$ conveys information on the effect a change in the j^{th} variable has. The standard error is employed to judge whether the effect of the j^{th} variable is distinguishable from 0 via a t-test (Montgomery and Peck, 1992) or a Wald test of the form $W = \frac{\hat{\beta}_j}{\text{se}_{\hat{\beta}_j}}$ (Krämer and Sonnberger, 1986). Using these two summary values per parameter thus recovers essential information about the original model. For more details on tests in linear

models and generalised linear models, please refer to Groß (2003) and McCullagh and Nelder (1989).

As M&R approach 1, I propose conducting the merge-step by taking the weighted mean for every summary value considered. The weights are chosen according to the number of observations $n_{b,i}$. Let $\underline{S}_{i-1}$ and $\underline{S}_i$ be the vectors of summary values for models $i-1$ and i that are to be merged. The new, merged vector of summary values $\underline{S}_{m\&r}$ is then obtained by

$$\underline{S}_{m\&r} = w_{i-1}\underline{S}_{i-1} + w_i\underline{S}_i, \tag{4.1}$$

where the weights w_{i-1} and w_i are given by $w_{i-1} = n_{b,i-1}/(n_{b,i-1} + n_{b,i})$ and $w_i = n_{b,i}/(n_{b,i-1} + n_{b,i})$, respectively. $n_{b,i-1}$ and $n_{b,i}$ are the number of observations the two models are based upon.

Such a merge-step ensures by construction that the complexity of merged models does not increase, the reduce-step is thus inherently included in the merge-step. The final model returns the specified summary values.

According to assumption (L1) on page 2, X is a non-stochastic matrix with rank p. I will now further assume that the values of X are realisations from the same arbitrary random distribution with expected value $-\infty < \underline{\mu}_X < \infty$ and variance $\underline{\sigma}^2_X < \infty$ and investigate some properties of the merging approach. Let us call this additional assumption (L5). For linear models, the standard least squares estimator $\underline{\hat{\beta}}$ has expected value $\mathsf{E}(\underline{\hat{\beta}}) = \underline{\beta}$ and variance-covariance-matrix $\mathsf{Cov}(\underline{\hat{\beta}}) = \sigma^2(X'X)^{-1}$ (Groß, 2003). As the estimator is unbiased regardless of X, for every block $b = 1, \ldots, B$, the expected value of the least squares estimator on that block is $\mathsf{E}(\underline{\hat{\beta}^b}) = \underline{\beta}$. Merging the models does not change the expected value, this approach thus results in an unbiased estimator for the true parameters $\underline{\beta}$.

The estimator's variance-covariance-matrix on the other hand does depend on X. I will now concentrate on the principal diagonal of this matrix, which contains the variances for every $\beta_j,\ j = 1, \ldots, p$. The square root of these values gives the standard errors. The variance for the j^{th} component is given by $\mathsf{Var}(\beta_j) = \sigma^2(X'_{\cdot j}X_{\cdot j})^{-1}$. Blockwise estimation of the linear models leads to a variance of $\mathsf{Var}(\beta_j^b) = \sigma^2(X^{b\,\prime}_{\cdot j}X^b_{\cdot j})^{-1}$ in block b.

The estimation of σ^2 does also vary, but $\hat{\sigma}^2$ is an unbiased estimator (Groß, 2003). For that reason, any systematic differences between $\mathsf{Var}(\beta_j)$ and its blockwise counterpart $\mathsf{Var}(\beta_j^b)$ are caused by the inverse of the respective version of $(X'_{\cdot j}X_{\cdot j})$. How does this change influence the variance of the estimator? According to assumptions (L1) and (L5), X is non-stochastic, but

its entries stem from an arbitrary distribution $(\underline{\mu}_X, \underline{\sigma}_X^2)$. As n grows, the diagonal elements of $X'X$ grow linearly in their expected value. The elements on the secondary diagonal on the other hand will grow slower than the elements on the primary diagonal in the case of $n \gg p$. Because of this, the main diagonal of the inverse $(X'X)^{-1}$ decreases approximately proportional to $\frac{1}{n}$ as n grows. This means in turn that the fraction $\frac{\mathsf{Var}(\beta_j^b)}{\mathsf{Var}(\beta_j)}$ is approximately equal to $\frac{n_b}{n}$. Because we merge the blocks by obtaining the weighted mean of the respective previous summary statistics, the estimation of the variance is not unbiased, rather, it is approximately $\frac{n_b}{n}$. Let us consider the following: For every block except the last one, the estimated variance is approximately $\frac{n}{n_b}$. For the last block, the estimated variance may be higher, as there probably are less observations in the last block. However, this block then will also have a lower weight. Taking this effect into account, the variance will be overestimated by a factor of $\lceil \frac{n}{n_b} \rceil$. To counter this overestimation, I propose correcting for it by dividing all summary values that represent estimated standard deviations by

$$\underline{S}_{m\&r}^{*\text{corr}} = \frac{\underline{S}_{m\&r}^{*}}{\sqrt{\lceil n/n_b \rceil}}, \tag{4.2}$$

where $\underline{S}_{m\&r}^{*}$ stands for the portion of $S_{m\&r}$ that represents standard errors.

This approach is useful for all cases where frequentist regression models are considered. The results are unbiased, but if left uncorrected, the approach would overestimate the estimator's variance and consequently the coefficient's estimated standard errors. This would lead to conservative decisions concerning the importance of variables. In the case of $n \gg p$, the factor by which the variance is overestimated is known to be close to $\lceil \frac{n}{n_b} \rceil$ and can thus be corrected for using Equation (4.2). A strength of this approach is its great flexibility. It can be employed for many frequentist regression models as it is based on the estimates of the regression coefficients and their standard errors.

4.3.2 Characteristics of the posterior distribution as summary values (M&R approach 2)

M&R approach 2 is aimed at conducting Bayesian regression analysis employing the Merge & Reduce-technique. In such a setting the aim is to characterise the posterior distribution $p(\underline{\beta}|X, \underline{Y})$ or its approximation by an MCMC-sample. The summary values should be chosen such that they characterise the distribution in a concise manner. More than one reasonable and useful solution is possible. Here, I utilise the mean and median, the upper and lower quartile, the 2.5%

and 97.5% quantile and the standard deviation of the MCMC-sample for every component β_j as summary values,

$$S = (\bar{x}_1, \ldots, \bar{x}_d, \tilde{x}_{u,1}, \ldots, \tilde{x}_{u,d}, s_1, \ldots, s_d),$$

where $u \in \{.025, .25, .5, .75, .975\}$ stands for the posterior quantiles considered, including the posterior median. This choice gives a good indication of the location and variation of the posterior distribution. However, for some purposes, other summary values may offer a better representation of the results. I recommend careful consideration of the requirements of one's analysis.

As in the frequentist case, when merging models, the number of observations the models are based upon are used as weights to ensure that every observation is of equal importance for the final model, confer Equation (4.1). For the Bayesian case, different correction factors are required. For measures of location, no correction is needed for symmetric posterior distributions. For the posterior standard deviation, I employ the same procedure as for the standard error in Section 4.3.1, i.e. divide the posterior standard error by $\sqrt{\left\lceil \frac{n}{n_b} \right\rceil}$. Other quantiles of the posterior distribution that can be interpreted as measure of dispersion such as the upper and lower quartile and the 2.5% and 97.5% quantile follow a slightly different pattern. In these cases, the distance between the quantiles and the measures of location grows with the ratio of the total number of observations and the number of observations per block $\frac{n}{n_b}$. Thus, a form of standardisation is necessary, resulting in

$$S_{u,j}^{\text{corr}} = \frac{S_{u,j} - \bar{x}_j}{\sqrt{\lceil n/n_b \rceil}} + \bar{x}_j. \tag{4.3}$$

For the quantiles that can be interpreted as a measure of dispersion, we thus subtract the posterior mean as a means of standardisation. We then divide the result by the correction factor $\sqrt{\lceil \frac{n}{n_b} \rceil}$ before adding the posterior mean again to arrive at the corrected posterior quantile.

4.3.3 Pointwise product (M&R approach 3)

M&R approach 3 presents a third way of merging two models by calculating their pointwise product. If the models can be represented by normal distributions f_1 and f_2, where $f_1 \sim N(\mu_1, \sigma_1^2)$ and $f_2 \sim N(\mu_2, \sigma_2^2)$, the pointwise product of the two distributions – indicated by a subscript P – is a normal distribution again, where the new parameters are

$$\sigma_P^2 = \frac{1}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} = \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 + \sigma_2^2}. \tag{4.4}$$

and

$$\mu_P = \left(\frac{\mu_1}{\sigma_1^2} + \frac{\mu_2}{\sigma_2^2}\right)\sigma_P^2. \tag{4.5}$$

In the multivariate case with $f_1 \sim N(\underline{\mu}_1, \Sigma_1)$ and $f_2 \sim N(\underline{\mu}_2, \Sigma_2)$, these equations change to

$$\Sigma_P^{-1} = \Sigma_1^{-1} + \Sigma_2^{-1} \tag{4.6}$$

and

$$\Sigma_P^{-1}\underline{\mu}_P = \Sigma_1^{-1}\underline{\mu}_1 + \Sigma_2^{-1}\underline{\mu}_2. \tag{4.7}$$

In the case of linear regression models, $\underline{\mu}$ stands for the expected value of the estimate $\underline{\hat{\beta}}$ and Σ_P stands for the variance of the estimator. Instead of saving $\underline{\mu}$ and Σ after every step, it is possible to recover the liner regression model by keeping $X_i'X_i$, $Y_i'\underline{X_i}$, $\underline{Y_i}'\underline{Y_i}$, and n_i. In the merge-step, these summary statistics are merged according to the following formulae:

$$\begin{aligned} X_m'X_m &= X_1'X_1 + X_2'X_2, & (4.8)\\ \underline{Y_m}'X_m &= \underline{Y_1}'X_1 + \underline{Y_2}'X_2, & (4.9)\\ \underline{Y_m}'\underline{Y_m} &= \underline{Y_1}'\underline{Y_1} + \underline{Y_2}'\underline{Y_2}, & (4.10)\\ n_m &= n_1 + n_2. & (4.11) \end{aligned}$$

After merging and reducing the final model, $X_m'X_m$ and $\underline{Y_m}'X_m$ are used to obtain the maximum-likelihood estimator $\underline{\hat{\beta}}$. This can easily be done by rewriting Equation (1.3) to

$$\underline{\hat{\beta}} = \left(X'X\right)^{-1}X'\underline{Y} = \left(X'X\right)^{-1}\left(\underline{Y}'X\right)', \tag{4.12}$$

using Equations (4.8) and (4.9). Equations (4.10) and (4.11) are employed to obtain the variance of the estimator. To that end, we first need an estimate of the model variance η^2. The maximum-likelihood estimator is given by

$$\hat{\sigma}_{\eta,ML} = \frac{\|\underline{Y} - X\underline{\hat{\beta}}\|^2}{n}.$$

However, $\hat{\sigma}_{\eta,ML}$ is a biased estimator. For that reason, the unbiased version

$$\hat{\sigma}_{\eta} = \frac{\|\underline{Y} - X\underline{\hat{\beta}}\|^2}{n-p}$$

can be preferable. In the following, I will employ $\hat{\sigma}_\eta$, but $\hat{\sigma}_{\eta,ML}$ can be used analogously. In both equations, it is necessary to calculate $\|\underline{Y} - X\underline{\hat{\beta}}\|^2$, which is given by

$$\|\underline{Y} - X\underline{\hat{\beta}}\|^2 = \underline{\hat{\beta}}' X' X \underline{\hat{\beta}} + \underline{Y}'\underline{Y} - 2\underline{Y}' X \underline{\hat{\beta}}.$$

From these components, the estimator's variance is given by

$$\mathsf{Var}(\underline{\hat{\beta}}) = \hat{\sigma}_\eta^2 \left(X_m' X_m\right)^{-1}. \tag{4.13}$$

In the merge-step of M&R approach 3, no explicit weighting is done during the merge-step. Instead, this role is assumed implicitly by Equations (4.8), (4.9), and (4.10).

M&R approach 3 is only suitable for cases where the result on each block can be expressed using normal distributions. This is the case for frequentist linear regression models and may be the case for some Bayesian linear regression models. In such cases, however, M&R approach 3 recovers the result on the full data set exactly. For that reason, no correction term is needed.

4.4 Generalised linear models

Generalised linear models (GLMs) offer a framework with more flexibility regarding the distribution of $\underline{Y}$. The roles of the design matrix X and the parameter vector $\underline{\beta}$ remain unchanged, likewise the assumptions concerning them. The link between $\underline{Y}$ and $X\underline{\beta}$ is now established employing a monotone link function g, which transforms the expected value of $\underline{Y}$ given $X\underline{\beta}$. This leads to Equation (4.14), the general formulation of a GLM:

$$g(\mathsf{E}(\underline{Y})) = X\underline{\beta}, \tag{4.14}$$

or, equivalently, expressed via the inverse $h(\underline{u}) = g^{-1}(\underline{u})$ (for $\underline{u} \in \mathbb{R}^n$):

$$\mathsf{E}(\underline{Y}) = h(X\underline{\beta}). \tag{4.15}$$

Special cases of a GLM are logistic regression, where the elements of $\underline{Y}$ are binary; Poisson regression, which is a standard model when $\underline{Y}$ contains count data; and linear regression, in which case the link is the identity function. Table 4.1 contains an overview of the three special cases of generalised linear models. A more detailed version of this table can be found in McCullagh and Nelder (1989, Chapter 2, Table 2.1).

Model	Type of data ($\underline{Y}$)	Canonical link	Link name	Alternatives
Linear regression	normally distributed	$g(\underline{u}) = \underline{u}$	identity	
Logistic regression	binary data	$g(\underline{u}) = \ln\left(\frac{\underline{u}}{\underline{1}-\underline{u}}\right)$	logit	probit, cloglog
Poisson regression	count data	$g(\underline{u}) = \ln(\underline{u})$	log	

Table 4.1: Some special cases of generalised linear models, the type of data and the link function. Abridged version of Table 2.1 from McCullagh and Nelder (1989).

It is possible to employ Merge & Reduce to conduct the analysis of GLMs on large data sets. To that end, I employ Poisson regression models both in the simulation study and in the analysis of the bicycle sharing data set in this chapter. Other types of GLMs require different structure in the data, but the structure of the result is similar and can be recovered well by M&R approaches 1 and 2 presented in Section 4.3.

When employing GLMs, some steps of the analysis change, among them the distribution of the dependent variable, the link function, significance testing, and regression analysis. This chapter does not provide an exhaustive list of differences between simple and generalised linear models. In the context of Merge & Reduce, it is important that a certain amount of structure of the result is retained for all types of models considered here and that the interpretation of the results can be done as it would have been done on the full model. Certain limitations apply, prominently, diagnostics are very difficult to perform. This issue is raised in Section 4.7 as well as Chapter 5.

4.5 Simulation study

The simulation study in this chapter is conducted for different statistical models: linear regression, linear regression with unmodelled mixtures, and Poisson regression. Both the data generation (Section 4.5.1) and the presentation of the results (Section 4.5.2) follow this structure of three different types of statistical models. The subsections of Section 4.5.2 are subdivided according to the corresponding M&R approaches.

4.5.1 Data generation

Linear regression The majority of data sets for the simulation study is generated along the same principle employed for the empirical evaluation of random projections (Subsection 2.4.1). For the linear case, the variables n, p and σ^2 are used. The number of observations n range from 20 000 to 1 000 000, the number of variables p from 5 to 200. The variance of the error term σ^2

takes values of 0.1, 1, 4, 25, 49, and 100, see also Table 4.2. A minority of data sets were created using function `mvrnorm` from R-package `MASS` (Venables and Ripley, 2002). For the models on these data sets, an additional setting was introduced: some models contain an intercept term while other do not.

I vary the number of observations per block n_b between relatively high number like 20 000 and 25 000, a range of high to medium-high numbers and very low number like 400. On modern computers, choosing values like $n_b = 20\,000$ or more usually does not pose any computational problems. I include values of n_b on the low side of the spectrum for two reasons. On the one hand, exploring possible limitations of the procedure is of interest. On the other hand, while the focus for the Merge & Reduce algorithm is on handling very large data sets on modern computers it may also be employed on computers or even sensors with low computational power. Reducing n_b gives an indication of what might happen in such an environment.

With the simulation study I try to systematically examine the influence of the parameters on the outcome. However, this does not mean that all possible combinations of n, p, σ^2, and n_b have been considered exactly once. Instead, some of the settings have been chosen up to five times to allow for an assessment of the variation that may happen even when the settings are kept constant. Other combinations have not been considered at all, an obvious example is the combination of $n = 20\,000$ and $n_b \geq 20\,000$.

parameter	role	values
n	number of observations	$\{20\,000, 50\,000, 100\,000, 500\,000, 1\,000\,000\}$
p	number of variables	$\{5, 50, 100, 200\}$
σ^2	error term variance	$\{0.1, 1, 4, 25, 49, 100\}$
n_b	number of observations per block	$\{400, 1\,000, 5\,000, 10\,000, 15\,000, 20\,000, 25\,000\}$

Table 4.2: Overview of parameters in simulation study.

Linear regression in the presence of unmodelled mixtures In general, there are two different types of unmodelled mixtures: mixtures of X and mixtures of both X and $\underline{\beta}$. In the first case, all assumptions of the linear model are fulfilled, the parameter vector $\underline{\beta}$ is valid for all values of X. This means that there will be systematic differences with respect to X and $\underline{Y}$ between the two groups. However, all observations follow the same linear model. If one group is small, their members are thus characterised by high leverage scores but inconspicuous Cook's distances. In the second case, the outliers will also follow a different regression model. Thus, the observations will not only show high leverage scores, but also high Cook's distances, as these

observations greatly influence the estimated model. This second situation is far from ideal from a modelling perspective and will not be considered here.

In the scenario where mixtures are present in the data set, the variables *proportion* and *pos* additionally come into play. Table 4.3 shows an overview of the two additional parameters and the values they take. The first variable, *proportion*, controls how much of the data set is made up of the undetected mixture. Values range from 0.01 to 0.5, but a majority of simulated data sets employ 0.05. The variable *pos* controls the position of the outlying values, I consider these four settings of *pos* in the simulation study:

`first` outliers first, followed by main body of observations;

`last` main body first, followed by outliers;

`middle` first half of main body, followed by outliers, followed by second half of main body;

`random` random arrangement.

The Merge & Reduce-technique is by construction invariant towards permutations of order of the blocks. The variable *pos* permutes the order of the observations. For that reason, the position of the mixture component may well have an influence on the results as it directly influences the proportion of the smaller mixture component in the blocks.

parameter	role	values
proportion	proportion of smaller mixture component	$\{0.01, 0.05, 0.1, 0.25, 0.5\}$
pos	position of smaller mixture component	{first, last, centre, random}

Table 4.3: Overview of parameters in simulation study for the scenario of mixtures on the level of X.

Poisson regression In the Poisson case, only the variables n and p from Table 4.2 are used when generating the data sets. Changes in X or $\underline{\beta}$ are not included in the analysis, *pos* and *proportion* are thus not used. The variable σ^2 also cannot be employed as the variance is equal to the mean value in the Poisson case by definition. Here, I choose the two parameters from $n \in \{50\,000, 100\,000\}$ and $p \in \{5, 10, 20\}$.

4.5.2 Results

In this section, I present the results of the simulation study. Section 4.5.2.1 describes the results of linear regression models, followed by results of unmodelled mixtures in Section 4.5.2.2 and of

Poisson regression models in Section 4.5.2.3. In each section, the results for each suitable M&R approach are presented in order.

In all of the sections, two different measures are considered to evaluate how well the M&R-models recover the original models. The first measure $e_m^2 \in \mathbb{R}_0^+$ is the squared Euclidean distance between an estimate for $\underline{\beta}$ based on M&R and another estimate based on the full data set:

$$e_m^2 = \|\hat{\underline{\beta}}_{MR}^m - \hat{\underline{\beta}}_{orig}^m\|_2^2, \quad m = 1, \ldots, M, \tag{4.16}$$

where m is the index of the simulated data set, $\hat{\underline{\beta}}_{MR}^m$ is the estimate for β for data set m based on one of the Merge & Reduce approaches, $\hat{\underline{\beta}}_{orig}^m$ is the respective estimate according to the original optimal model and M is the total number of simulated data sets considered in the current setting. Instead of estimates for $\underline{\beta}$ as in Equation (4.16), I analogously employ posterior means, posterior medians or (corrected) posterior quantiles as well. A distance of 0 means that the results returned by the model on the full data set and the M&R approach are exactly identical. Large values on the other hand indicate great differences between the two results. In the following, I arbitrarily choose a cut-off of 0.1 to stand for a good approximation.

The second measure $f_{\text{se}}^m \in \mathbb{R}_0^+$ is suitable to evaluate variances or standard deviations. It assesses how well the standard deviation is recovered by the M&R approaches and can be interpreted as a factor. The corrected standard error factor is defined as

$$f_{\text{se}}^m = \frac{1}{p} \sum_{j=1}^{p} \frac{\text{se}_{j,MR}^m}{\text{se}_{j,orig}^m}, \tag{4.17}$$

where $\text{se}_{j,orig}^m$ and $\text{se}_{j,MR}^m$ are the (corrected) estimated standard errors for variable j, $j = 1, \ldots, p$ according to the maximum likelihood estimate and the Merge & Reduce approaches. When $f_{\text{se}}^m = 1$, the standard errors are the same for both models. Values greater than 1 indicate that the M&R approach inflates the standard errors while values less than 1 stand for a reduction of the standard errors. In all following sections, I consider values in the interval $[0.975, 1.025]$ to stand for an acceptable approximation. Calculating the mean of all individual j standard error factors in Equation (4.17) may be surprising, but in my simulation studies I found that all these factors were identical up to at least 5 digits.

4.5.2.1 Results for linear regression

M&R approach 1 For data sets which were generated following a linear model, all three merging strategies introduced in 4.3 are applicable and suitable. M&R approach 1 – keeping the

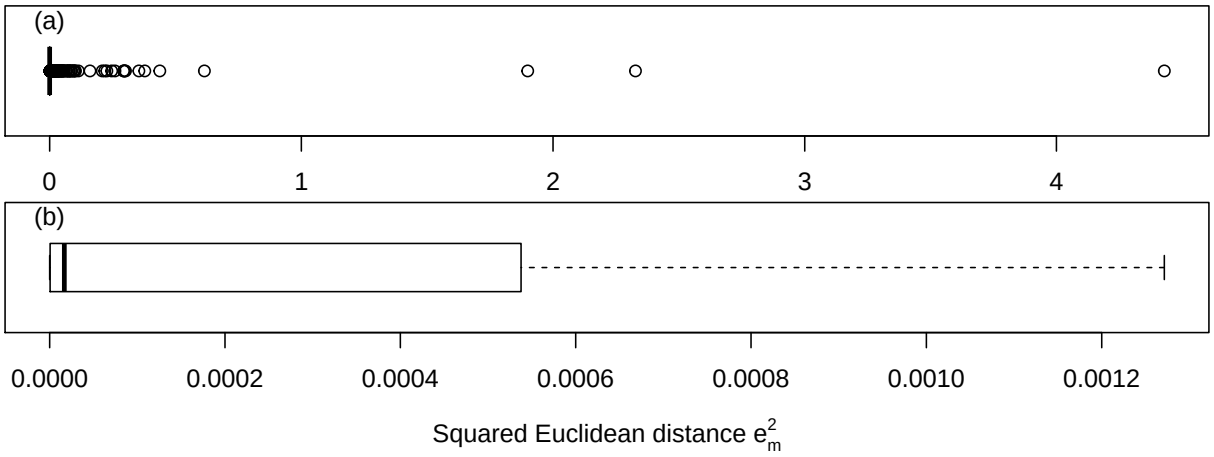

Figure 4.2: Boxplots of squared Euclidean distances between M&R approach 1 and original model observed in the simulation study, across all values of n, p, σ_ε, and n_b. Subfigure (a) contains all values, subfigure (b) excludes 180 out of 952 values that lie further than 1.5 times the interquartile range away from the upper quantile.

estimate and the standard error as summary values – is useful for frequentist linear regression or for Bayesian cases where the posterior distribution can be obtained analytically. In the following I will quantify how well it recovers the original estimator $\hat{\underline{\beta}}_{orig}$ – which is often the Gauß-Markov estimator – by employing e_m^2 (Equation (4.16)). Figure 4.2 depicts these values e_m^2. Here, $M = 952$.

Subfigure (a) of Figure 4.2 shows all values of e_m^2 observed in the simulation study across all parameter settings, including different values for the number of variables p, which adds summands to the Euclidean distance. The box on the extreme left of the plot covers only a small range of values, but there are 172 outliers whose majority are close to 0.1 while three simulation settings result in values of e_m^2 of just under 2, around 2.5 and above 4. Subfigure (b) leaves out the outliers and shows only the box and whiskers of the squared distances. Here we can see that 75% of all values obtained in the simulation studies lie between 0 and 0.0006. The median of e_m^2 takes the very low value of 0.000022. All values above 0.0016 are considered outliers in the boxplot as they are further away from the box than the interquartile range multiplied with 1.5. All of this indicates a highly right-skewed empirical distribution of the squared Euclidean distances e_m^2.

A further look at quantiles reinforces this impression. Table 4.4 shows selected quantiles of e_m^2 rounded to four decimal places. Again, we can see that for the vast majority of settings in the simulation study Merge & Reduce recovers the results of the original linear model with high accuracy, for almost all of the settings, the squared Euclidean distance is less than 0.1, for many, it is even lower.

Quantile	min.	50%	75%	90%	95%	97.5%	99 %	max.
Squared distance	0.0000	0.0000	0.0006	0.0109	0.0355	0.0870	0.2956	4.4287

Table 4.4: Selected quantiles of the observed squared Euclidean distances between parameter estimates following M&R approach 1 and the original linear model.

In the next step, I examine which of the settings produce the high outlying values. In Figure 4.2, Subfigure (a), we can see three distinct outliers which are settings that lead to squared Euclidean distances e_m^2 greater than 1. For these three cases the settings are $\sigma^2 = 100$, $p = 400$, and $n_b = 200$. There are two more simulated data sets with these settings. For these, the squared Euclidean distances are also relatively high with values of 0.3 and 0.35, respectively. In general, cases whose associated distance is relatively high comprise at least one of the following settings: high error variance, a high number of variables or a low number of observations per block.

To reiterate this, we may encounter deviations from the original linear model if we split a data set with a large number of variables into comparatively small blocks, especially if the variance of the error term is high. This combination poses a difficult situation for a linear model. Harrell (2001) recommends a ratio of $\frac{n}{p} \geq 10$ or even $\frac{n}{p} \geq 20$ if the response variable is continuous. Models with a lower ratio of $\frac{n}{p}$ cannot be expected to be reliable according to Harrell, i.e. their predictive power for new observations may be low. The results of our simulation study suggest that similar or slightly higher ratios of observations per variable are required to employ M&R approach 1.

Further investigation indicates that the ratio n_b/p is indeed the deciding factor, confer Figure 4.3. We observe the highest values of e_m^2 if $n_b/p = 2$, the smallest value this ratio takes in our simulation study. The distances decrease by a large amount for the next lowest ratio, $n_b/p = 4$. The simulation study suggests that for values of $4 \leq n_b/p \leq 10$, the squared Euclidean distances may occasionally be high, while for $10 < n_b/p \leq 25$, some squared distances higher than 0.1 may be observed. For $n_b/p > 25$, no deviation in our simulation study was above 0.1. The error term variance σ^2 seems to play a secondary role as high values of σ^2 make higher squared distances between M&R model and original model more likely. However, even for the highest error term variance considered ($\sigma^2 = 100$), we observe very low squared distances regularly. The total number of observations in the data set, n, does not seem to influence the distance between original model and M&R model.

It is interesting to note that e_m^2 is dominated by the distance between the estimated intercept in many cases. For the majority of simulation settings, this fraction is higher than 0.8 or even

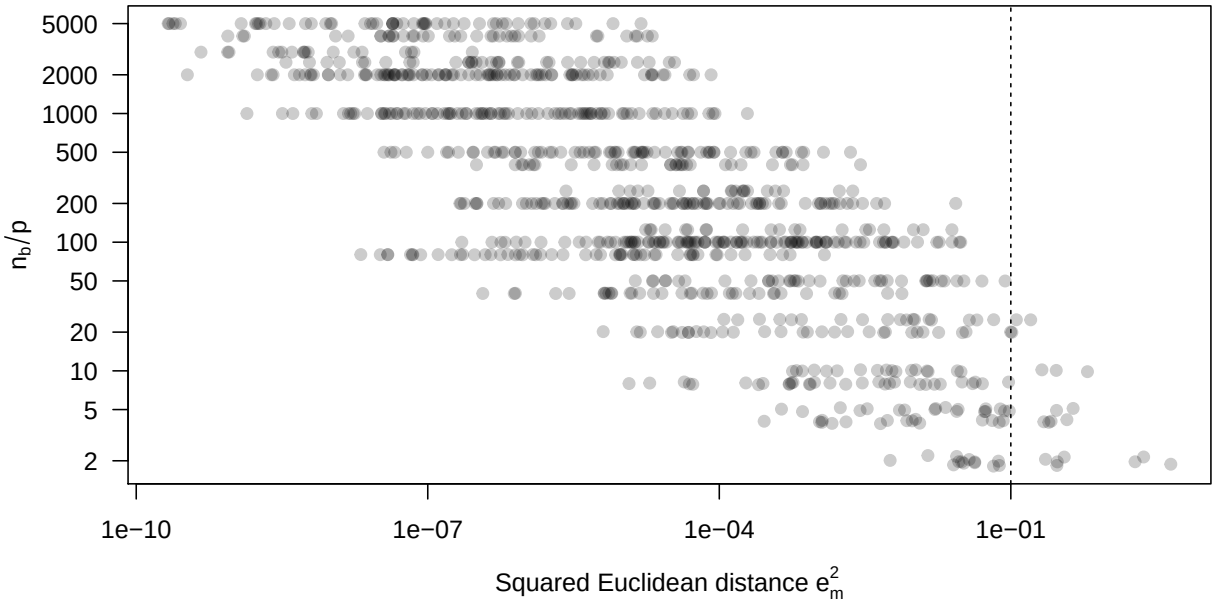

Figure 4.3: Scatterplot of the effect of observations per block per variable $\frac{n_b}{p}$ on squared Euclidean distances $e_m^2,\ m = 1, \ldots, M$ for M&R approach 1. x- and y-axes are on a logarithmic scale, observations are drawn as partially transparent points: gray points mean single observations, black points multiple observations at roughly the same location. Vertical dashed line is at 0.1.

0.9. This is in particular true for the cases where e_m^2 is exceptionally high. Surprisingly, the intercept's influence on e_m^2 seems to be largely independent of p. Even for simulated data sets with 200 variables, the fraction of e_m^2 caused by the intercept lies above 0.8 for most of the data sets. Ignoring the intercept and calculating e_m^2 employing only the estimates for all other variables results in very low numbers with a maximum of 0.107.

After the difference in the estimators I now assess the estimated standard errors. In contrast to the estimated values $\underline{\hat{\beta}}$, the corresponding estimated standard errors behave similarly for intercept and all other variables. For that reason, I will report the corrected standard error factor f_{se}^m (Equation (4.17)).

Subfigure (a) of Figure 4.4 shows the corrected values f_{se}^m for all settings in the simulation study. The kernel density estimate of the corrected standard error factors shows a clear peak around 1. However, there are still relatively high values of $f_{\text{se}}^m > 1.4$ for some settings, which means that the estimated standard error is inflated by more than 40%. To gain a better insight into when this happens, I will look at f_{se}^m split according to $\frac{n_b}{p}$.

Subfigure (b) of Figure 4.4 shows the effect of the blocksize n_b in comparison to the number of variables p. There is an obvious connection between $\frac{n_b}{p}$ and f_{se}^m: for relatively high values of $\frac{n_b}{p}$, the correction works well and results in values of f_{se}^m close to 1. As $\frac{n_b}{p}$ decreases, f_{se}^m increases. There are some values of f_{se}^m below 1. This is the case when the ratio $\frac{n}{n_b}$ is not integer valued,

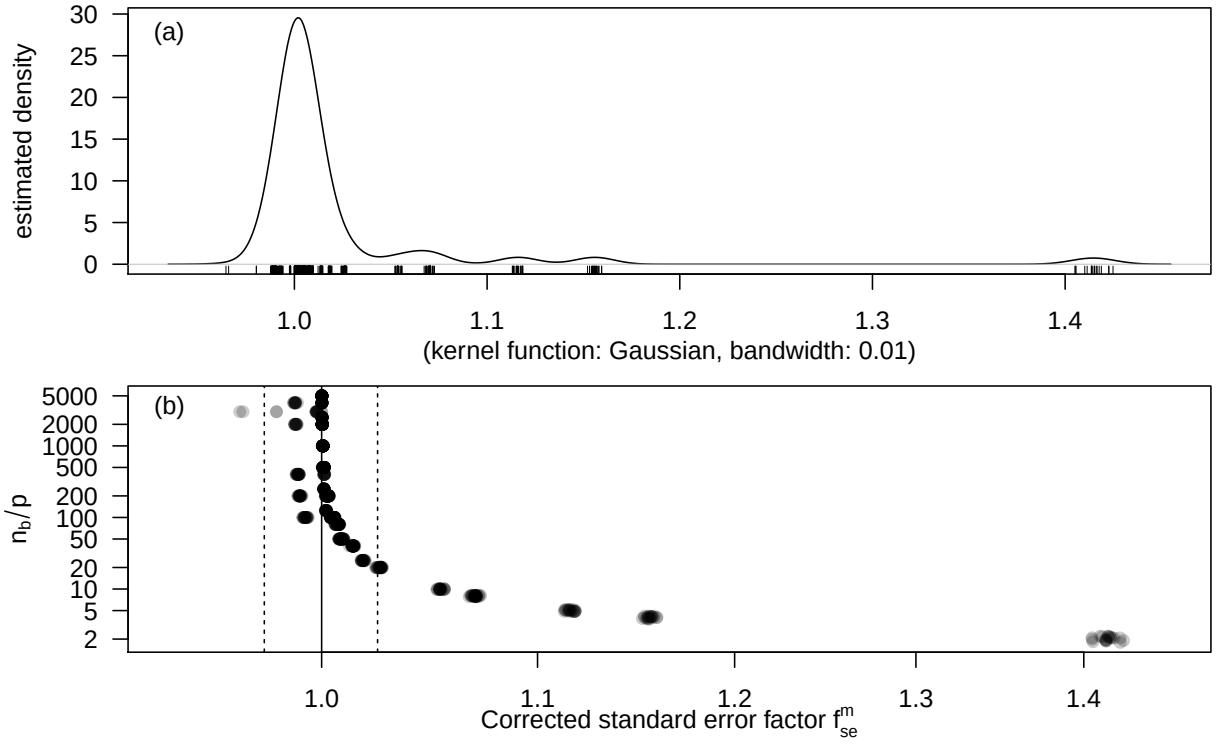

Figure 4.4: Display of the corrected standard error factors f_{se}^m across all simulated settings for M&R approach 1. Subfigure (a) shows a kernel density estimate, subfigure (b) shows a scatterplot of the effect of observations per block per variable $\frac{n_b}{p}$ on corrected standard error factors f_{se}^m across all settings for M&R approach 1. In subfigure (b), the y-axis is on a logarithmic scale and observations are drawn as partially transparent points: grey points mean single observations, black points multiple observations at roughly the same location. The solid vertical line indicates values of $f_{\text{se}}^m = 1$, the two dotted lines stand for relative deviations of 2.5%, i.e. $f_{\text{se}}^m = 0.975$ and $f_{\text{se}}^m = 1.025$.

e.g. $n = 50\,000$ and $n_b = 20\,000$. In such cases, dividing by $\left\lceil \sqrt{\frac{n}{n_b}} \right\rceil$ is a slight overcorrection. Variables will thus tend to look slightly more important and p-values of the associated t-tests for significance will be slightly lower than in the original model. However, the values of f_{se}^m in question are very close to 1, a minimum of $\min_m f_{\text{se}}^m = 0.9876$ means that estimated standard errors are 1.24% off in comparison to the original model.

Similarly to the result of the analysis of the squared distance between estimators, e_m^2, we again see that the approximation quality of the corrected estimated standard errors, measured by the factor f_{se}^m, depends on $\frac{n_b}{p}$. For values of $\frac{n_b}{p} \leq 10$, the estimated standard errors are inflated by a factor of between 5% and 40%. If $\frac{n_b}{p} \geq 20$, the inflation is generally less then 2.5%, which we consider an acceptable value. In case of the estimated standard error, both intercept and other variables behave virtually identically with a uniform inflation factor applying to every entry. The error term variance σ^2 on the other hand does not seem to have any influence on

f_{se}^m. The settings of $\sigma^2 = 0.1$ and $\sigma^2 = 49$ have only been included twice each in the simulation study, but the four other settings have featured in more than 200 data sets per setting and show virtually identical behaviour.

To conclude the frequentist case, both estimated effects $\underline{\hat{\beta}}$ and estimated standard errors $\underline{\text{se}}_{\text{orig}}$ of the original model are recovered well when using the corrected standard errors $\underline{\text{se}}_{\text{MR}}$, provided the number of observations per block is high enough for the number of variables. I recommend values of at least $\frac{n_b}{p} \geq 20$. If this fraction is lower, the estimated standard errors may become inflated while the estimated intercept may be comparably far away from the intercept estimated by the original model.

M&R approach 2 In the Bayesian case, I utilise the same simulated data sets as for the frequentist case, but change the model accordingly. As explicated in Section 4.3.2, there are many possibilities of how to characterise the MCMC-sample of the posterior distribution. In the following, I will present results for measures of location as they are returned by the Merge & Reduce-algorithm. Posterior quantiles other than the median are corrected for the effect of the ratio $\frac{n}{n_b}$ employing Equation (4.3). The corrected posterior standard deviation is obtained using Equation (4.2).

Figure 4.5 shows two boxplots, similarly to Figure 4.2 in the frequentist case. Here, the squared Euclidean distance e_m^2 between the posterior medians according to Merge & Reduce approach 2 and the original Bayesian model are considered across all settings in the simulation study, resulting in a total of 786 observations. As in the frequentist case, three values of $e_m^2 > 1$ can be found, with the highest value slightly below 4. Subfigure (b) indicates that some more variation between original model and Merge & Reduce model is present compared to the frequentist case, however, with 75% of all values of e_m^2 lower than 0.0013, the differences are very small for the majority of all results from the simulation study. The 95% quantile of e_m^2 is 0.0473. Results for the posterior mean are very similar.

Figure 4.6 shows a plot containing the squared Euclidean distances between the posterior median based on M&R approach 2 and the original Bayesian model for all simulated data sets grouped according to the ratio $\frac{n_b}{p}$. The distances exhibit the same pattern we observed for M&R approach 1, i.e. the median is well recovered by M&R approach 2 provided $\frac{n_b}{p} > 25$, but can be unreliable for smaller ratios of $\frac{n_b}{p}$. Results for the posterior mean are almost identical.

In the next step, I will consider the posterior standard deviations and their characteristics in both the Bayesian linear models and their Merge & Reduce counterparts. Analogously to

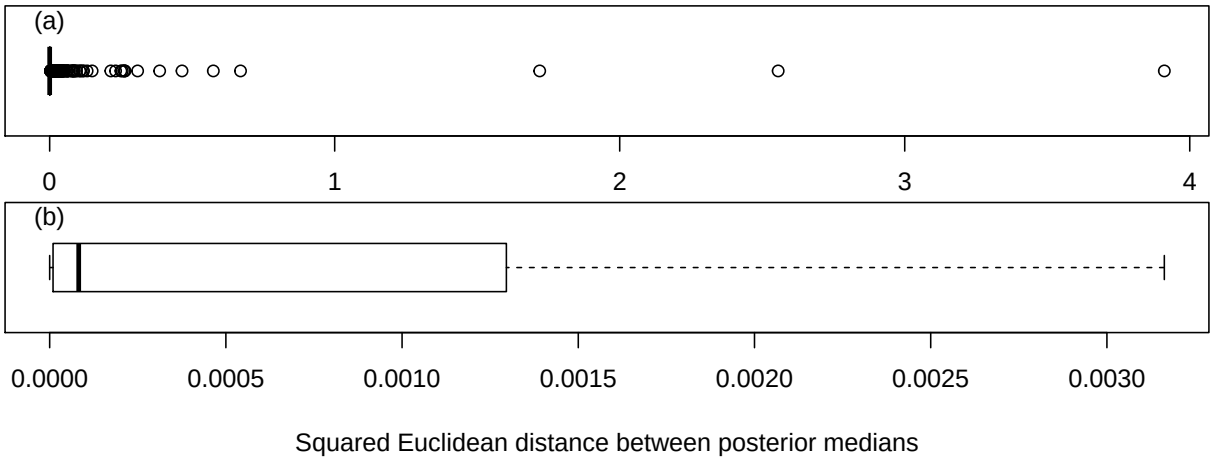

Figure 4.5: Squared Euclidean distances $e_m^2,\ m = 1, \ldots, M$ between posterior medians according to M&R approach 2 and original Bayesian model observed in a simulation study, across all values of n, p, σ, and n_b. Subfigure (a) contains all values, subfigure (b) excludes 155 out of 786 values that are considered outliers.

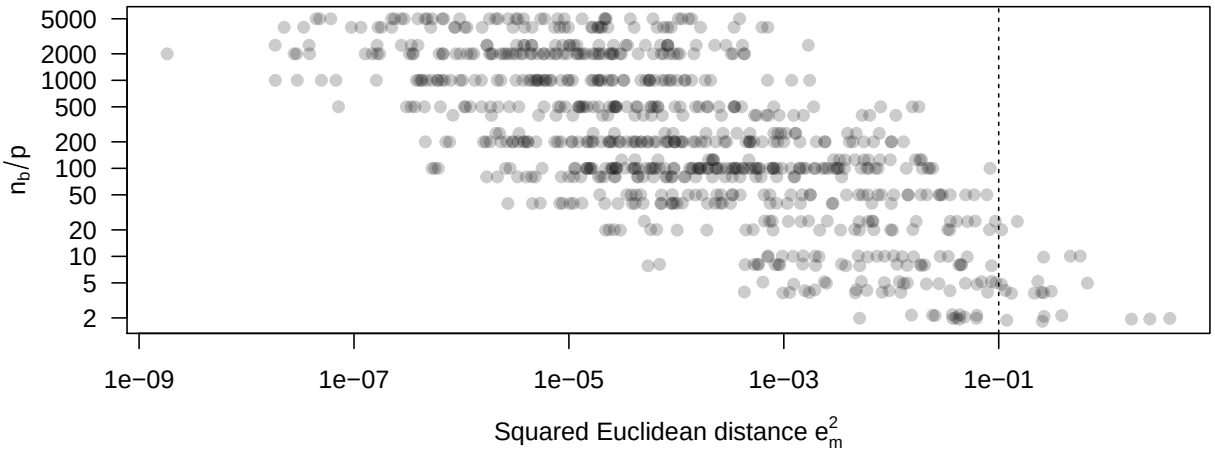

Figure 4.6: Scatterplot of the effect of observations per block per variable $\frac{n_b}{p}$ on squared Euclidean distances $e_m^2,\ m = 1, \ldots, M$ between posterior medians for M&R approach 2. x- and y-axes are drawn on a logarithmic scale, observations are drawn as partially transparent points: grey points mean single observations, black points multiple observations at roughly the same location. Vertical dashed line is at 0.1.

the frequentist case, I employ the mean of the quotients of the corrected posterior standard deviations according to M&R approach 2 and the posterior standard deviations according to the original model as in Equation (4.17). These values f_{se}^m are shown in Figure 4.7, grouped by both n_b and p. If we again consider values of f_{se}^m in the interval $[0.975, 1.025]$ as acceptable, almost all parameter settings of n_b and p with the exception of $n_b = 5\,000$ exhibit unacceptably high or low values, with $n_b = 10\,000$ coming close to leading to acceptable values only. In contrast to the results for frequentist linear regression, there are highly inflated posterior standard deviations

even for a high number of observations per block and a low number of variables. In total, the posterior standard deviation does not seem to be a useful and reliable summary value in the context of Merge & Reduce.

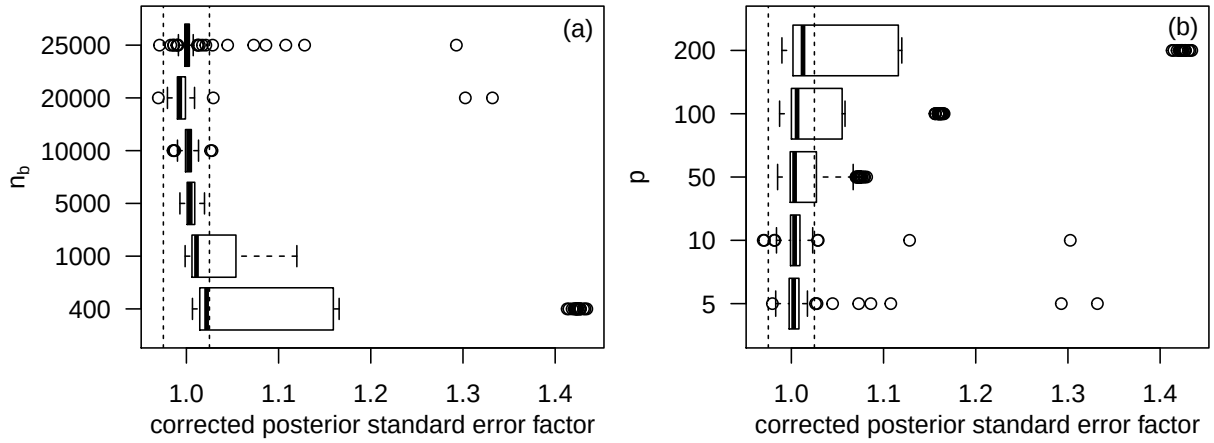

Figure 4.7: Boxplots of corrected standard error factor f_{se}^m. Subfigure (a) shows the effect of block size n_b on f_{se}^m, subfigure (b) the effect of the number of variables p on f_{se}^m. The two dotted lines stand for relative deviations of 2.5%, i.e. $f_{\text{se}}^m = 0.975$ and $f_{\text{se}}^m = 1.025$.

Instead, I will consider the corrected versions of the posterior quartiles as well as the posterior 2.5% and 97.5% quantiles. For the correction, I employ Equation (4.3), which takes into account that while measures of location are well-recovered, the distance of other quantiles to mean or median seems to be grow proportionally to $\sqrt{\lceil \frac{n}{n_b} \rceil}$.

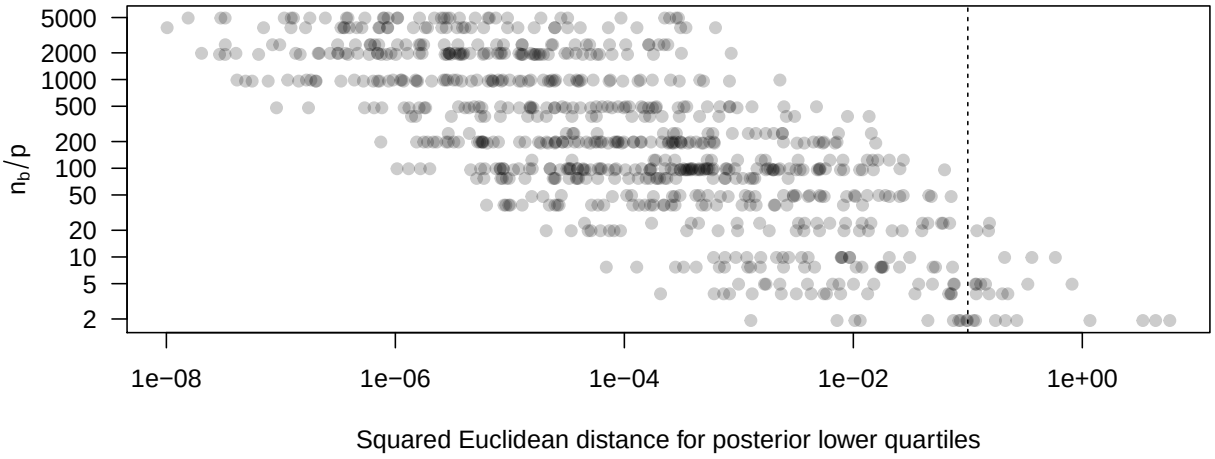

Figure 4.8: Scatterplot of the effect of observations per block per variable $\frac{n_b}{p}$ on squared Euclidean distances between posterior lower quartiles for M&R approach 2. Both x- and y-axis are on a logarithmic scale, observations are drawn as partially transparent points: grey points mean single observations, black points multiple observations at roughly the same location. Vertical dashed line is at 0.1.

Figures 4.8 and 4.9 show the squared Euclidean distances between posterior lower quantiles and posterior 97.5% quantiles, respectively. The results show the same patterns found in the comparison of posterior means and medians: the ratio of observations per block and variables plays an important role. For low values of $\frac{n_b}{p}$, the distances between the original model and the M&R result may become too high. As $\frac{n_b}{p}$ grows, these distances tend to take lower values. For $\frac{n_b}{p} \geq 50$, all M&R models in the simulation study at hand are able to recover the posterior distribution of the original Bayesian linear model very well, even for the 2.5% and 97.5% quantiles.

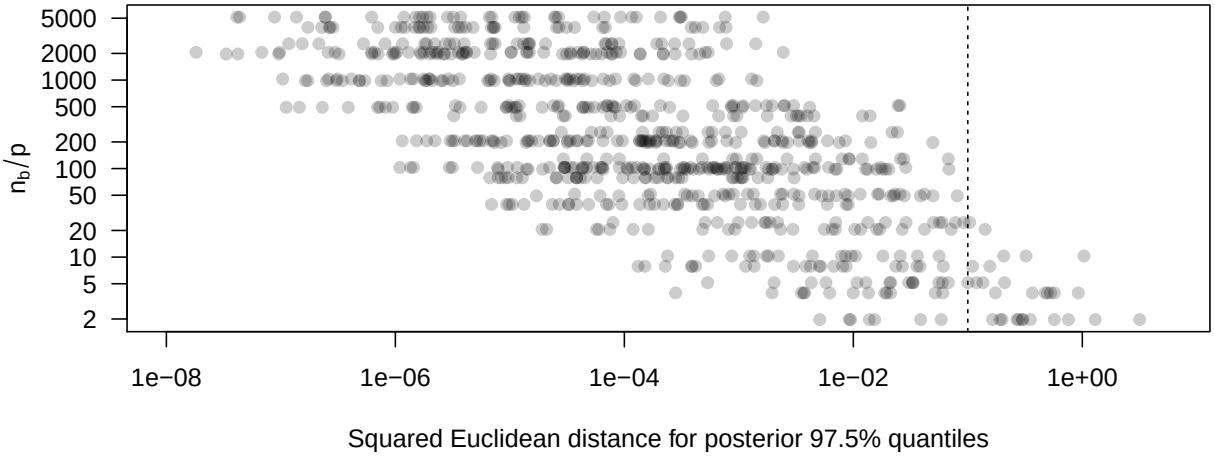

Figure 4.9: Scatterplot of the effect of observations per block per variable $\frac{n_b}{p}$ on squared Euclidean distances between posterior 97.5% quantiles for M&R approach 2. Both x- and y-axis are on a logarithmic scale, observations are drawn as partially transparent points: grey points mean single observations, black points multiple observations at roughly the same location. Vertical dashed line is at 0.1.

M&R approach 3 M&R approach 3 works very well for the case when all assumptions are met. All squared Euclidean distances e_m^2 are 0 for $m = 1, \ldots, M$, which indicates that $\underline{\beta}_{MR}^m$ and $\underline{\beta}_{orig}^m$ are identical up to numerical precision of the machine for all simulated data sets. The standard error factors f_{se}^m is 1 up to at least 8 digits. M&R approach 3 is thus able to recover the exact results of the original frequentist linear regression model up to numerical precision. For the blockwise analysis of frequentist linear regression models, M&R approach 3 thus is the best choice.

4.5.2.2 Results for linear regression with unmodelled mixtures

In this section, I will examine the second scenario of the simulation study: the presence of unmodelled mixtures. The simulated data sets contain outlying observations that differ in the X-values and possibly also in the values of $\underline{Y}$. However, they follow the same underlying linear

model. Here, all simulated data sets have been chosen such that the ratio $\frac{n_b}{p}$ is greater than 50 to avoid problems encountered in Section 4.5.2.1. In this approach, I only consider frequentist regression models. For that reason, I will limit the analysis to M&R approaches 1 and 3, as M&R approach 2 is suitable for Bayesian regression models.

In this section, I will analyse the influence of the position of the outliers. To that end, the simulated data sets contain four different ways of merging the two subsets. The respective setting is recorded in the variable *pos*, which can take the values `first`, `last`, `middle`, and `random`. Section 4.5.1 contains a more detailed explanation of the different settings.

As mentioned in Section 4.5.1, the principle of Merge & Reduce is constructed in a way that ensures the order of the blocks does not influence the result. Changing the position of the mixture component is not the same as changing the order of the blocks, mainly because the simulated data sets are created according to the same principles but do not necessarily contain the same observations, but also, because changing the order of observations is generally not equal to changing the order of blocks. The values of *pos* lead to different degrees of homogeneity in the blocks. Random arrangement leads to blocks that consist of a small portion of outliers where the expected relative frequency of outliers is constant for all blocks. The other values of *pos* on the other hand lead to a majority of blocks with only observations from the main body and a small number of blocks with a high relative frequency of outliers or possibly even blocks consisting of outliers only. For that reason, we can expect very similar behaviour for data sets that were created with the settings `first` and `last`, similar behaviour for the setting `middle` but possibly different behaviour for the setting `random`.

M&R approach 1 Figure 4.10 shows the squared Euclidean distances e_m^2 (confer Equation (4.16)) for all simulated data sets, split by the position of the outliers. We can clearly see that parameter estimates are well-recovered regardless of the position of the outliers. The largest squared distance has a value of 0.112 and is thus only slightly above our arbitrarily set boundary of 0.1. The position of the outliers clearly holds an influence on the values of e_m^2, settings where the outliers are randomly inserted into the data seem to be systematically better at recovering the original model's results. Data sets where the outliers are either the first or the last observations exhibit the highest values of e_m^2 in comparison.

Figure 4.11 shows the corrected standard error factor f_{se}^m (confer Equation (4.17)) for all simulated data sets, split by the position of the outliers. Here, the influence of the outliers' position becomes even more obvious. This is plausible as blocks that only contain observations from one

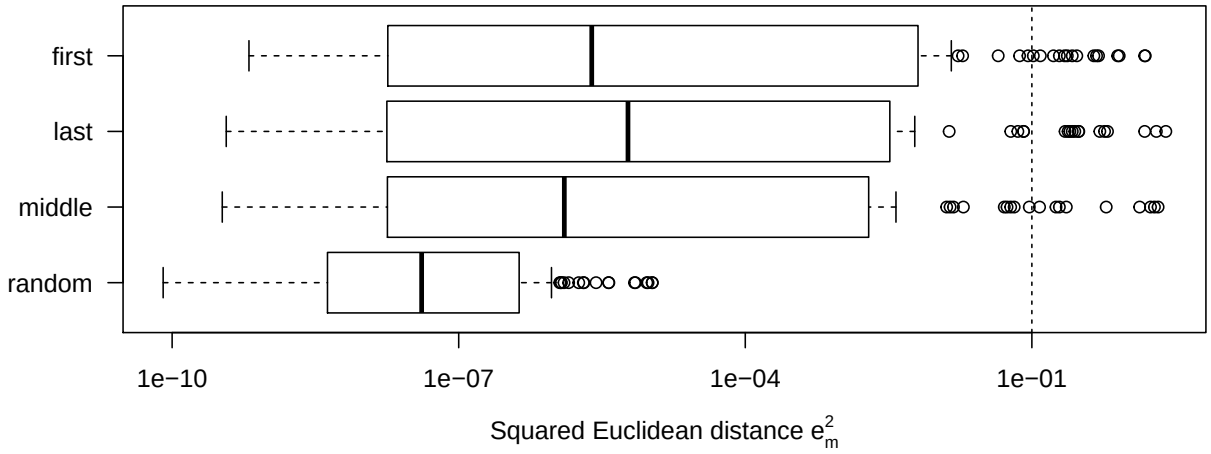

Figure 4.10: Boxplots of the effect of position of the outlying observations on squared Euclidean distance e_m^2 in mixture scenario for M&R approach 1. y-axis is on a logarithmic scale, vertical dashed line is at 0.1.

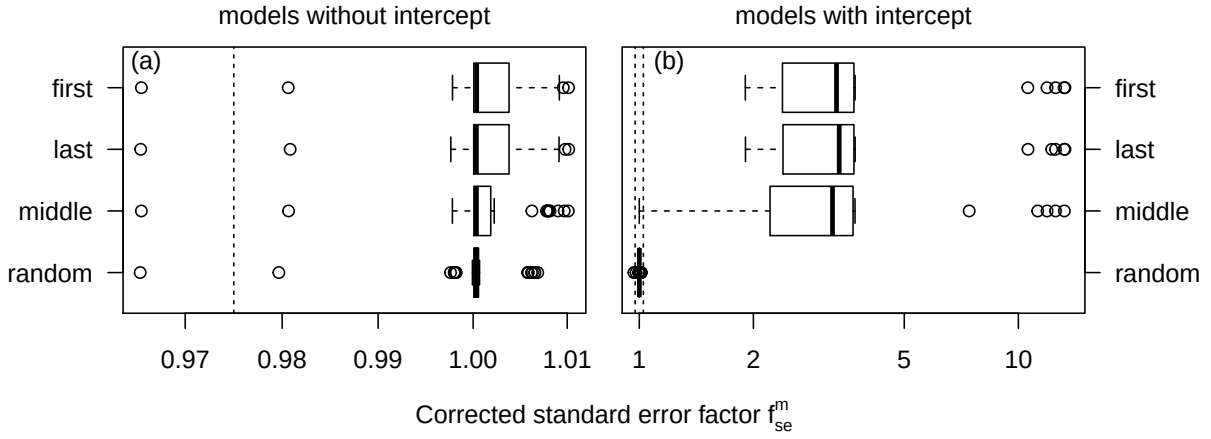

Figure 4.11: Boxplots of the effect of position of the outlying observations on standard error factor f_{se}^m in mixture scenario for M&R approach 1. Subfigure (a) contains models and data sets without intercept term, subfigure (b) contains models and data sets with an intercept term. y-axis is on a logarithmic scale, vertical dashed lines are at 0.975 and 1.025.

of the mixture components necessarily underestimate the variation in the data, influencing the estimated standard error.

When considering the corrected standard error factor, whether the model contains an intercept term or not has considerable consequences. Subfigure (a) of Figure 4.11 shows all models without an intercept term. Random assignment and outlying observations in the centre of the data set lead to the smallest values of f_{se}^m, but for all settings in the simulation study without an intercept, f_{se}^m is below 1.025, representing a minor inflation of the standard error.

In subfigure (b) on the right-hand side, we see all models with an intercept term. For data sets with random arrangement of the outliers, the estimated standard errors are as well-recovered

by the Merge & Reduce-technique as the ones without an intercept term. For data sets where all outliers appear together as a group, the standard errors are systematically overestimated by the Merge & Reduce-solution.

In conclusion, M&R approach 1 is able to recover the parameter estimates of the original models well in all cases considered in our simulation study. The estimated standard error is only well-recovered if the model contains no intercept term or all outliers are randomly allocated across the data set. The position of the outliers generally plays a role for the ability to recover the original model's results. This reflects the increased difficulty of merging heterogeneous blocks that are based on homogeneous data. The results also indicate that the order of the blocks does not influence the results, indicated by the high similarity between results for simulated data sets that contain outlying observations in the beginning of the data set and those that contain outlying observations at the end of the data set.

M&R approach 3 M&R approach 3 again is able to recover the original linear model virtually perfectly. As in Subsection 4.5.2.1, all squared Euclidean distances e_m^2 are 0 up to the numerical precision of the machine and all standard error factors f_{se}^m are 1 up to 8 digits. M&R approach 3 is thus neither influenced by permutations in the order of the observations nor in the presence of a group of outliers.

In cases of frequentist linear regression, M&R approach 3 thus provides a very useful and appropriate approach as it leads to exactly the same result a linear regression on the full data set gives, even if the data comes from two different sources, that do not change the regression model, but influence the location of the observations.

4.5.2.3 Results for Poisson regression

As last element of the simulation study, I now consider data sets where the dependent variable $\underline{Y}$ follows a Poisson distribution. Fittingly, I analyse these data sets using a Poisson regression model. The following simulation study is on a smaller scale than the previous ones, containing a total of six data sets with values of $n = \{50\,000, 100\,000\}$, $p = \{5, 10, 20\}$, and $n_b = \{400, 1000, 5000, 10\,000, 20\,000, 25\,000\}$. The error term standard deviation σ cannot be chosen here as the variance is equal to the mean for Poisson-distributed data by definition. M&R approach 3 is not included in this section. To evaluate how well M&R approaches 1 and 2 are able to recover the results of the original Poisson model, I again employ the squared Euclidean distances e_m^2 (Equation (4.16)) for the parameter estimates for M&R approach 1 as well as for

all summary values for M&R approach 2. The corrected standard error factor (confer Equation (4.17)) is used for the standard error for M&R approach 1 and additionally for the posterior standard deviation for M&R approach 2.

M&R approach 1 Figure 4.12 shows the squared Euclidean distances for all Poisson models in the simulation study. We can clearly see that all distances take very low values, even for the lower values of observations per block per variable, which are $\frac{n_b}{p} = 20$ for the Poisson regression models. The estimates of all Poisson models are thus well-recovered by M&R approach 1.

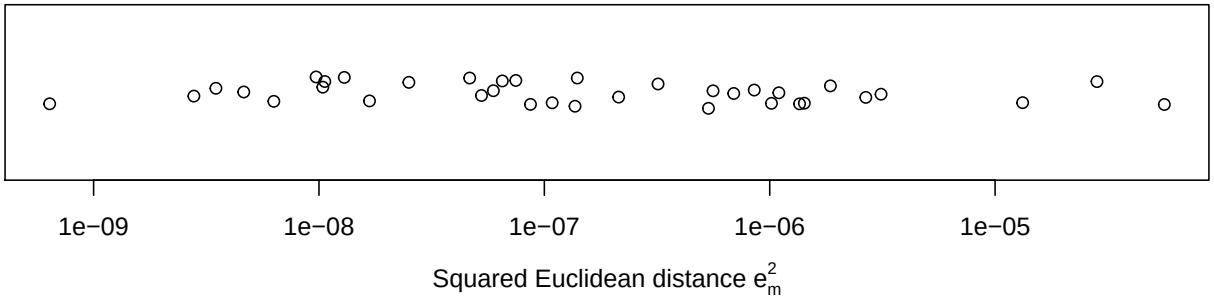

Figure 4.12: Stripchart (one-dimensional scatterplot) of squared Euclidean distances e_m^2 for all Poisson regression models for M&R approach 1. x-axis is on a logarithmic scale. Vertical dashed line at 0.1 is not visible due to small values of e_m^2.

Figure 4.13 shows the corrected standard error factor. Here, all but two values are within the acceptable range of $[0.975, 1.025]$. The two values outside of that range come from simulated data sets with $n_b = 400$ and $p = 20$, indicating that the standard error factor exhibits a greater dependence on $\frac{n_b}{p}$ than the parameter estimate. The settings that lead to the next highest values of f_{se}^m come from simulated data sets with $\frac{n_b}{p} = 40$. They lead to comparatively high values of f_{se}^m around 1.015, well within our interval of acceptable values.

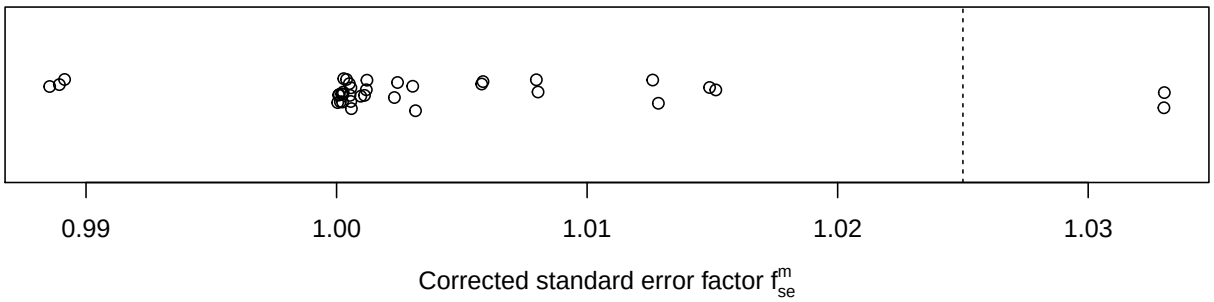

Figure 4.13: Stripchart of corrected standard error factors f_{se}^m for all Poisson regression models for M&R approach 1. Vertical dashed line is at 1.025.

M&R approach 2 To evaluate the performance of M&R approach 2 for Bayesian Poisson regression models, we first look at the squared Euclidean distances e_m^2 between the original posterior means and the posterior means based on Merge & Reduce. Figure 4.14 shows a one-dimensional scatterplot of the distances. All values are very small, the largest squared distance lies below 0.0001. The posterior location is thus well-recovered by M&R approach 2.

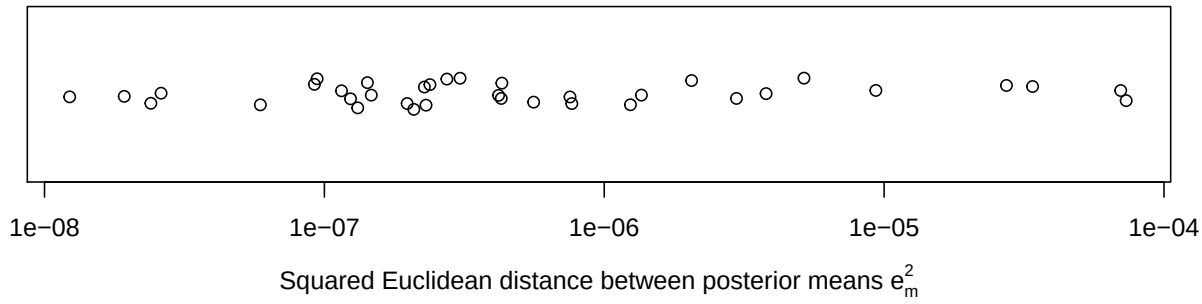

Figure 4.14: Stripchart of squared Euclidean distances between posterior means of original model and Merge & Reduce model for all Bayesian Poisson regression models for M&R approach 2. x-axis is on a logarithmic scale. Vertical dashed line at 0.1 is not visible due to small values of e_m^2.

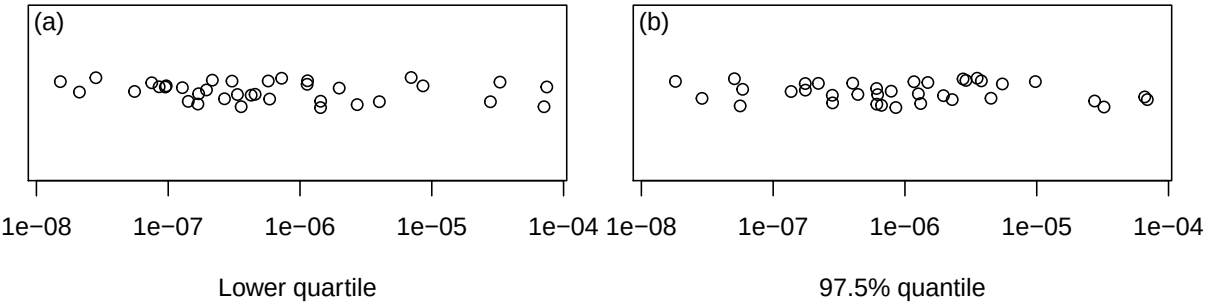

Figure 4.15: Stripchart of squared Euclidean distances between posterior 25% and 97.5% quantiles of original model and Merge & Reduce model for all Bayesian Poisson regression models for M&R approach 2. x-axes are on a logarithmic scale. Vertical dashed line at 0.1 is not visible due to small values of e_m^2.

To characterise the posterior distribution, in addition to measures of location, we again look at posterior quantiles. I do not include the posterior standard deviation as it performed poorly for Bayesian linear regression models. Figure 4.15 contains the squared Euclidean distances for the lower quartile and the 97.5% quantile as representatives. The quantiles based on M&R approach 2 are corrected according to Equation (4.3). With the correction, all squared distances obtain very low values, again, all lie below 0.0001. The Poisson regression models are thus well-recovered for both frequentist and Bayesian models.

	Quantitative variables only				Including factor variables			
	logarithmic		Poisson		logarithmic		Poisson	
n_b	e^2	f_{se}	e^2	f_{se}	e^2	f_{se}	e^2	f_{se}
10 000	0.0676	1.0078	0.0443	1.0184	0.1595	0.9412	0.0988	1.0139
5000	0.0796	1.0279	0.0316	1.0574	0.0898	0.9193	0.0288	1.0182
1000	2.6690	1.4216	0.3666	1.5174	1.9845	0.9399	0.7687	1.1529
400	5.9530	1.5078	0.9500	1.6528	10.2661	1.0118	7.7288	1.2610

Table 4.5: Results of the frequentist analyses of the bicycle sharing data set. Each row headed with e^2 gives the squared Euclidean distance between the original model and the model obtained using M&R approach 1 with the given block size n_b, while each row headed with a f_{se} shows the corrected standard error factor. A total of four models are analysed: two models using only quantitative variables as independent variables and two models including factor variables as well. For each of these models, a linear regression and a Poisson regression analysis are conducted.

4.6 Bicycle rental data set

For evaluation on a real data set, I employ the bicycle rental data set introduced in Section 2.5. In contrast to Chapters 2 and 3, I use a different set of variables from the data set. This is done because the bicycle rental data set contains factor variables that appear in a very systematic way, e.g., the variable *yr*, which indicates whether the observation belongs to the first or second year under study. For this variable, there is one change from 2011 to 2012 in the middle of the data set, but otherwise the variable stays constant. This introduces problems with identifiability when employing Merge & Reduce as for the majority of blocks not all of the values are present. This variable's effect on the number of bicycle rentals can thus not be estimated for most blocks.

For that reason, I consider two models, one where only the quantitative independent variables *atemp*, *hum*, and *windspeed* remain in the model and a second model similar to the model used in Section 2.5, but without the variables *yr* and *season*. For both models, I analyse both a linear and a Poisson regression model. In the linear case, the dependent variable is transformed using the natural logarithm.

The original data set contains $n = 17\,379$ observations, I thus employ block sizes of $n_b \in \{400, 1000, 5000, 10\,000\}$. Table 4.5 gives an overview of how close the results according to Merge & Reduce are to the original model for the frequentist analysis while Table 4.6 shows the results for the Bayesian analysis.

The results are very similar for both approaches (again representing frequentist and Bayesian models, respectively), for both linear and Poisson regression, and for both sets of variables: the approximation of the original model by the models based on Merge & Reduce is good for block sizes $n_b \in \{5000, 10\,000\}$ but bad for the two smaller block sizes.

variab.	model	n_b	$\tilde{x}_{0.025}$	$\tilde{x}_{0.25}$	$\bar{x}$	$\tilde{x}_{0.5}$	$\tilde{x}_{0.75}$	$\tilde{x}_{0.975}$	s
quant.	log.	10 000	0.0683	0.0683	0.0678	0.0676	0.0672	0.0665	1.0016
quant.	log.	5000	0.0783	0.0792	0.0810	0.0809	0.0834	0.0872	1.0148
quant.	log.	1000	2.4053	2.5722	2.6725	2.6761	2.7800	2.9838	1.3290
quant.	log.	400	5.4856	5.7779	5.9472	5.9437	6.1249	6.4711	1.3973
quant.	Pois.	10 000	0.0445	0.0443	0.0443	0.0443	0.0442	0.0442	1.0262
quant.	Pois.	5000	0.0315	0.0315	0.0316	0.0316	0.0317	0.0317	1.0634
quant.	Pois.	1000	0.3592	0.3641	0.3667	0.3667	0.3694	0.3742	1.5349
quant.	Pois.	400	0.9368	0.9455	0.9502	0.9503	0.9549	0.9635	1.6705
factors	log.	10 000	0.1674	0.1636	0.1619	0.1622	0.1608	0.1579	0.9298
factors	log.	5000	0.0999	0.0953	0.0932	0.0933	0.0915	0.0873	0.9128
factors	Pois.	10 000	0.0988	0.0988	0.0989	0.0988	0.0989	0.0991	1.0208
factors	Pois.	5000	0.0291	0.0289	0.0288	0.0288	0.0287	0.0286	1.0141

Table 4.6: Results of the Bayesian analyses of the bicycle sharing data set. A total of four models are analysed: two models using only quantitative variables as independent variables and two models including factor variables as well. For each of these models, a linear regression (using a logarithmic transformation of the number of shared bikes as dependent variable) and a Poisson regression analysis are conducted. Every row shows the squared Euclidean distances e_m^2 between the original model and the model obtained using M&R approach 2 the respective block size n_b for one of the four models. The approximations of different posterior quantiles $\tilde{x}$ as well as posterior mean and standard deviation are given. The models with factor variables and block sizes $n_b \leq 1000$ did not converge, resulting in meaningless large deviations from the original model. For that reason, they are not included in the table.

While the results are similar across all settings, a difference can be seen between the model including only the three quantitative variables and the model which includes factor variables as well. Especially in the Bayesian case, the model with factors shows large deviations from the original model when the block size is small. On closer inspection, this is mainly due to the variable *holiday*, which is not present in all of the blocks and thus leads to divergent models where at least some elements of the posterior distribution of $\underline{\beta}$ are meaningless. I walked into this trap even though I excluded two variables because of their unbalancedness. This illustrates the importance of careful model selection with possibly little information especially when factor variables are present. It also underlines that n_b should not be chosen too small.

As mentioned in Section 4.5.2.1, Harrell (2001) recommends a ratio of at least 20 observations per variable for a reliable linear model, i.e. $\frac{n}{p} \geq 20$. Harrell uses the effective number of observations m instead of n. For models that include only quantitative variables, $m = n$, but if factor variables are included in the model, the effective number of observations changes. Let k be the number of factor levels and let n_i be the number of observations for level i, $i = 1, \ldots, k$. The effective number of observations m is then given by $m = \min(n_1, n_2)$ for binary variables and by $m = n - \frac{1}{n^2} \sum_{i=1}^{k} n_i^3$ for factor variables with $k > 2$ levels (see Harrell, 2001, Table 4.1).

Block size n_b	quantitative variables only		including factor variables	
	$\min n_{\text{eff}}$	$\frac{\min n_{\text{eff}}}{p}$	$\min n_{\text{eff}}$	$\frac{\min n_{\text{eff}}}{p}$
400	179	44.75	0	0
1000	379	94.75	0	0
5000	2379	594.75	95	2.64
10 000	7379	1844.75	191	5.31

Table 4.7: Smallest value of effective number of observations m for different block sizes n_b as well as minimal effective number of observations divided by number of parameters p, where $p = 36$ including all dummy variables.

I now calculate m for all blocks and for all four values of n_b. For the model that only includes quantitative variables, m is constant except for the last block. For the model with factors, m will typically vary as the frequency of the different levels varies across blocks. Table 4.7 reports the smallest number of m observed across all blocks with the respective value of n_b as well as the resulting minimal number of observations per variable.

From Table 4.7 it becomes obvious that 400 and 1000 are not adequate block sizes for the bicycle rental data set and the model that includes the factor variables. For $n_b = 400$, around half of all blocks do not contain a holiday. For $n_b = 1000$, this is only the case for one block. However, even the blocks that do contain a holiday typically include only one holiday, giving an effective number of observations of 24, which is less than the number of variables. This emphasises that care should be taken especially when possibly unbalanced binary variables are present in the model.

While not as extreme, the results for the smaller model that only contains the three quantitative variables *atemp*, *hum*, and *windspeed* show the same pattern: acceptable approximation for block sizes 5000 and 10 000, but high deviations for block sizes 400 and 1000. Including the intercept term, the minimal number of observations per variable is 44.75 and 94.75 for the two problematic block sizes, respectively. The number of observations per variable is thus higher than required for a good approximation as observed in the ideal situation of the simulation study.

In combination, these results indicate that the effective number of observations per variable is one important factor to recover the results well. On the other hand the Merge & Reduce-technique is able to recover the more complex model including factor variables well even for ratios $\frac{\min m}{p}$ that are lower than ratios for models with three quantitative variables that could not be approximated well. In addition to the number of observations per variable, the model's goodness-of-fit – which is considerably higher for the more complex model – seems to also play a role.

As mentioned before, I propose employing Merge & Reduce as a method to conduct regression analysis on very large data sets that possibly are too large to feasibly be analysed on the whole. In such a setting, choosing a value of n_b that is large enough should not be a problem – on the contrary, very small block sizes would be impractical.

4.7 Conclusion

Out novel Merge & Reduce-technique on statistical models works well for a variety of regression models. For frequentist linear regression, M&R approach 3 offers a way of recovering the original model with virtually no difference using point-wise products of normal distributions, even if outliers with regard to X and $\underline{Y}$ are present that follow the same regression model.

For frequentist linear regression and other frequentist regression models, both the parameter estimates and their estimated standard errors are well-recovered by M&R approach 1 provided the ratio of observations per block and variables is high enough. The parameter estimates can be employed directly following the approach described in Section 4.3.1, while the estimated standard errors can be merged in the same way, but have to be corrected using $\sqrt{\lceil \frac{n}{n_b} \rceil}$ (confer Equation (4.2)). The approach was evaluated on linear and Poisson regression models and works well for both. It seems plausible that M&R approach 1 can also be employed for regression models with a similar structure like other GLMs.

For data sets that contain outliers that still follow the same linear model, the parameter estimates are also well-recovered. In such a situation, however, the estimated standard errors are overestimated unless the model does not contain an intercept term or the outliers are distributed relatively evenly across the data set.

For Bayesian regression models and M&R approach 2, the results regarding measures of location like mean and median are comparable – provided there are enough observations per variable in each block, the results are very close to those of the original model. For Poisson models, the corrected posterior standard deviation of the models according to M&R approach 2 is close to the posterior standard deviation of the original models, but this does not seem to be the case for the linear models. Quantiles that can be used as a measure of dispersion pose an alternative. After a correction based on the factor $\sqrt{\lceil \frac{n}{n_b} \rceil}$ (confer Equation (4.3)), these values are close the quantiles stemming from the original models.

Application of the technique to the bicycle rental data set highlights two important points. Firstly, factor variables can be problematic for Merge & Reduce, especially if they are unbalanced,

thus greatly reducing the effective number of observations in each block. Secondly, for the real data set, even Merge & Reduce models with a ratio of $\frac{n_b}{p}$ close to 100 – which seems to be more than enough under the ideal conditions of the simulation study – do not guarantee good approximation of the original model. The goodness-of-fit of the original model seems to play a role, models that do not explain the data well seem to require a higher ratio of $\frac{n_b}{p}$. It seems prudent to choose the block size n_b rather on the large side in order to avoid bad approximations of the original model. In a Big Data setting, this does not constitute a serious constriction.

In the frequentist case, the summary values are straightforward and contain all important information about the model. In the Bayesian case, there are a variety of suitable summary values derived from the MCMC-sample of the posterior distribution. In the thesis at hand, I have covered and examined some quantiles that may be useful to characterise posterior distributions. However, other interesting cases are not necessarily covered by this. I will mention two examples.

For extreme value statistics, extreme quantiles such as a 99% or 99.9% quantile may be of interest. In the simulation study, there difference between results for the 75% and the 97.5% quantiles is very low. This may indicate that generalisations to more extreme quantiles are possible, but it is by no means a guarantee.

In the simulation study at hand, all posterior distributions are symmetric or reasonably close. Models or data sets that lead to asymmetric posterior distribution pose an interesting challenge. On the one hand, it would be interesting to find summary values that can be utilised to describe the skewness. On the other hand, it is unclear whether the correction factor for the quantiles still works or at least needs to be adjusted as it treats the quantiles symmetrically in its current form.

A challenge I have not discussed in the current chapter is the inclusion of prior information. All models in both the simulation study and the analysis of the bicycle rental data set employed non-informative prior distributions. Including informative prior distributions in the Merge & Reduce-scheme is not trivial. As the number of observations per block may be considerably smaller than the total number of observations, it would presumably be necessary to make the prior distribution less informative for each block. What amount of information should remain in the prior distribution and how this aim could be reached are open research questions.

The Merge & Reduce-technique offers great flexibility and stability and seems to be useful for a broad range of regression models. However, checking assumptions and performing diagnostics on the model are difficult or impossible. This is necessarily the case as every block of data is deleted as soon as the respective model is built, there is thus no easy way of obtaining residuals

without a hefty increase of memory requirements. Some basic ideas to solve this are discussed in Section 5.1.3.

Chapter 5

Discussion, Open Problems, and Outlook

Statistically speaking you miss one hundred percent of the shots you don't take.

(inspired by Wayne Gretzky)

In this thesis, I have introduced and proposed two different approaches with the aim of making regression analysis on very large data sets possible.

5.1 Limitations of the proposed approaches

5.1.1 Focus on large amount of observations, not large amount of variables

The focus of this thesis lies on cases where the data sets contain a lot of observations n, but not a lot of variables p and where the aim is to reduce the number of observations while approximating the original model well. The approaches in Chapters 2, 3, and 4 are thus introduced for data sets where $n \gg p$.

Arguably, variable selection is the more prevalent task in current research. Can these approaches also be employed in cases where $n \ll p$?

For Merge & Reduce, this is not easily possible. As the number of variables becomes the problematic dimension, each block would hold information about the same n people, but would contain only a subset of the variables. Even worse, all the subsets would be disjoint. This would lead to great difficulties as we would have to merge models with disjoint set of parameters. Even before that, models might fit the data very poorly depending on the subset of variables.

For random projections, the situation looks a bit more friendly. In principle, it is possible to employ our techniques to a data set with $n \ll p$ by just transposing the data set and treating it as before. This would, however, lead to challenges with the interpretation of the results. Instead of the original variables, the projected data set contains random linear combinations of the original variables. It may be possible to find interpretations for these linear combinations, as e.g. in the case of principal component analysis. However, whereas principle component analysis is designed to find structure in the data and connections between the variables, the linear combinations in a projection are indeed random and built without any knowledge of the data. For that reason, applying random projections to reduce the number of variables is only sensible if the result is useful even though the variables cannot be interpreted.

Still, random projections or techniques based on the Johnson-Lindenstrauss-theorem in general can be useful for variable selection as well. Another option is to employ sampling techniques instead of random projections. Sampling approaches present a broad area of research in Statistics as well as Computer Science. While details lie beyond the scope of this thesis, the general idea is to sample according to weights that capture the structure in the data. One example is importance sampling where the weights are based on the norm of the observation ($n \gg p$) or the variable ($n \ll p$). In some cases, selected entities need to be adjusted in order to avoid a bias introduced by the sampling strategy.

Classically, sampling approaches had to be conducted in at least two steps – one step that determines the weights for each entity and a second step to sample the observations according to the weights. Each step is connected to one pass through the data. This is usually not practical in a streaming setting. However, there are now approaches that allow combining both steps, thus enabling a one-pass sampling algorithm.

Coresets are similar to sampling approaches. Again, the aim is to obtain a subset that approximates the data set.

5.1.2 Approximation for similar number of observations and variables

Related to the challenges discussed in Section 5.1.1 is the problem of a number of variables p that is less than the number of observations n, but not necessarily substantially less. This affects the approaches introduced in the thesis at hand in different ways.

For random projections, this does not pose a fundamental problem. They can be employed to reduce a data set to any number of observations k with $k > p$. Reducing the number of observations below the rank p of the data set is not advisable as the approximation may become

arbitrarily bad. For Merge & Reduce, it would in principle be enough to ensure that the block size n_b is greater than p. However, as we have seen in Section 4.5.2.1 and other sections of Chapter 4, in order for the linear model to be estimated reliably, it is vital that the data set contains at least 20 observations per variable for data sets consisting of quantitative variables. This constraint should be taken into account when reducing data sets.

For random projections, a second aspect plays a role. When employing random projections for linear regression models, a theoretical guarantee concerning the goodness of the approximation can be given. This guarantee requires the practitioner to choose an approximation parameter ε which is then used to calculate the number of observations k that is required to guarantee the goodness of approximation or to choose the desired number of observations k, which can then be used to calculate the implied approximation guarantee ε. As mentioned in Chapter 2, the relationship between ε and k does not depend on n, but it does depend on p. For different sketching methods, it may rely linearly or quadratically on p. The latter case implies that the required number of observations k very quickly increases as p grows – or, seen from another perspective, the approximation guarantee becomes less strong as p increases, unless n and also k increase at a suitable speed as well.

5.1.3 Diagnostics

Both random projections and Merge & Reduce enable analysing regression models on very large data sets. While both approaches employ the observations to obtain an approximation of the desired regression model, the observations are not readily available afterwards. In the case of Merge & Reduce, the observations are read in blockwise and are deleted as soon as the model on the respective block is built. This means, the observations would have to be read again, which may be infeasible or even impossible.

When using random projections, on the other hand, the sketched observations are available. These, however, represent random linear projections of the original observations. Each sketched observation incorporates multiple original observations with different weights by construction. In general, the sketched observations are not useful to identify possible problems in the model. Different approaches with a similar aim like importance sampling or coresets may pose an alternative here as they reduce n by sampling and possibly weighting observations from the original data set according to some criterion.

To be able to conduct diagnostics in spite of the unavailability of observations, one useful idea might be to uniformly sample a reasonable number of observations from the original data

set. In general, this may lead to biased or unreliable results if, e.g. a small group of influential observations is present that will be missed in uniform sampling with high probability. However, as we already have a reliable approximation of the original model, we can check whether the model based on the uniform sample is close enough to our approximated model and only conduct diagnostics on the sample if this is the case.

This very basic idea seems plausible, but it may be problematic to determine when the two models are 'close enough'. It also may not be helpful if simple random sampling is only able to capture the structure of the problem with low probability.

In general, the techniques introduced in the present thesis aim at approximating regression models, not at approximating the observations the models are based upon. In order to enable diagnostics as well as to a lesser extent predictions, it is an interesting open question how to find a good subset of the observations, especially in a setting where the data set cannot be kept in memory as a whole or in a streaming setting where every observations might only be read once.

5.2 Bachelor's and Master's theses

Over the course of this project, a number of Bachelor's and Master's theses have originated from the project. Two of the Master's theses – Rathjens (2015) and Müller (2016) – form the basis for Chapter 3. In this section, I give an overview of further final theses I have co-supervised that add to insights described in the thesis at hand.

Niels Lategahn's Master's thesis (Lategahn, 2016) is based on the assumption that all values of $\underline{Y}$ are unknown but observable, where uncovering each observation is linked to a high cost. Examples for this are experiments where the object of interest is destroyed or immaterial costs like invasive procedures for patients. It is further assumed that our budget allows for a subset of k observations. The aim is to choose the subset without prior knowledge of any values of $\underline{Y}$ and obtain a regression model that approximates the original model as good as possible.

Different algorithms to obtain the subsets are compared. The main idea is to ensure the subset represents the entirety of the full data set well. To that end, three main ideas are employed: clustering, sampling according to leverage scores, and ensuring the Euclidean distance between selected observations is maximal or does not become too low. Some approaches also consist of combinations of two principles. If appropriate, the observations in the subset are weighted according to the number of original observations they represent. All approaches are compared to a random sample of size k. Random sampling generally works well if all assumptions are met,

but can fail catastrophically if a good approximation of the model relies on selecting members of a relatively small group. To put the methods to test and to work out differences between the more sophisticated approaches, different problems or even violations of assumptions are included. These include outliers in the values of X or $\underline{Y}$, heteroscedasticity, and transformations and interactions that are not taken into account in the model. Please refer to Appendix B for a more detailed list of the violations considered.

The results indicate that a method inspired by Feldman et al. (2011) and a combination of k-means clustering and leverage scores perform best. In the first case, a two-step procedure is iteratively applied repeatedly. The observation with the highest leverage score is chosen, then a fixed number of observations closest to the chosen observation is removed from the pool. In the second case, k clusters are identified using k-means clustering. In each cluster, the observation with the highest leverage score is chosen as representative. In some scenarios, relying on maximising the distance between the selected observations shows advantages, but in general, it seems to be advantageous to take both the distance between observations and the importance for the regression model into account. Leverage scores are an important tool in this setting as they can be calculated without knowledge of $\underline{Y}$.

Alexander Wollenberg's Master's thesis (Wollenberg, 2016) deals with a reduction of large or high-dimensional data sets in the context of logic regression. Logic regression employs logic trees as independent variables (Schwender and Ickstadt, 2008). Such trees can be of the form A or $(B \wedge \neg C)$, where A, B, and C are logical expressions that can either be true or false. Employing logic trees and logic regression allows for the modelling of interactions of higher order, which is especially useful in a SNP-setting or in other contexts where redundancy or other causes for interactions play a role. The results can also be used to estimate variable importance (Schwender et al., 2011).

Finding the logic trees is not an efficient process and requires large amounts of computing power. For that reason, the size of the data sets that can feasibly be analysed using logic regression is limited. The aim of Wollenberg (2016) is to first sample from the problematically large dimension to filter out entities that contribute little or nothing to the regression model. This can be done for data sets with a large number of observations or a large number of variables, but variable reduction usually is the more interesting case for practitioners.

Especially for the case of high-dimensional data sets with many variables, sampling according to the leverage scores adds little value compared to simple uniform sampling, as there is considerable overlap between the distributions of the leverage scores for important variables and for

negligible variables. However, this is different for the cross-leverage scores between the independent variables $\underline{X}_i$ and the dependent variable $\underline{Y}$. Here, the distribution of important variables differs markedly from the distribution observed for negligible variables. This difference can be used by sampling variables according to their cross-leverage score with the dependent variable. No theoretical analysis has taken place, but empirical studies indicate that such a sampling procedure is indeed able to recover the important variables while discarding the negligible ones with high probability. This is true for both main effects and variables that are important via an interaction. Employed as a pre-processing step before the logic regression, this approach enables applying logic regression to data sets of considerably higher dimension.

Peter Gnändinger's Master's thesis (Gnändinger, 2018) applies a Bayesian logistic regression model on data on penalties taken in the German Bundesliga in association football. This model is an extension of Bornkamp et al. (2009). While the earlier work of Bornkamp et al. (2009) modelled the goalkeeper effect and took the logarithm of the total number of penalties by each penalty taker, Gnändinger (2018) additionally includes the effect of both goalkeepers and penalty takers. Interactions between goalkeepers and penalty takers can also be included in the model.

Jonas Münch's Bachelor's thesis (Münch, 2016) deals with large data sets, but not with regression models. The thesis is based on positional data of every player and the ball for two matches in the German Bundesliga in association football. For all players, the position is measured in two dimensions, for the ball, a third dimension is added. Every position is measured 25 times per second, which results in 135 000 observations per player not including stoppage time. The task is to find suitable visualisations that show the spatial distribution of a player's position over the course of a match. This can easily be adapted to separate positions on attack from positions on defence or positions up until the first goal and positions after the first goal. The distribution can be visualised as a heat map with some modifications – e.g. overlaying the pitch with a honeycomb structure – or its mean or median can be given. The latter is advantageous if the focus is not on one player but on the team as a whole.

In a research project, we expanded the visualisations to also include positions where the ball is won from the opposing team. It is then possible to interactively follow the next sequence to see what the team and its opposition made of this opportunity. Back to static displays, we tried to characterise where teams play the ball after having won possession. In ongoing research in cooperation with TU Dortmund University's School of Journalism and Mass Communication,

we investigate to what extend such visualisations aid in understanding reports about a football match.

5.3 Conclusion

In this thesis, I introduce two approaches that can be employed when conducting regression analysis on very large data sets. Both approaches focus on data sets with a very large number of observations n and a considerably smaller number of variables p, $n \gg p$.

Random projections have been developed from the Johnson-Lindenstrauss-theorem. In the case presented here, they reduce the number of observations from n to k with $p < k < n$ while capturing the structure of the regression problem. Inserting a projection step before the regression analysis allows for efficient modelling of regression problems even for very large data sets. Such solutions already existed for frequentist linear regression. They have been extended to Bayesian linear regression by Geppert et al. (2017) and to more generalised regression model in further work. With this approach, the focus lies on a good approximation of the regression model using an embedded data set instead of the original one. Random projections offer theoretical guarantees for the approximation. The required number of observations in the embedded data set k only depends on the number of variables p and the approximation parameter ε, in particular, the required number is independent of n. This makes the approach very useful for very large data sets with a small to moderate number of variables. The theoretical guarantees hold for frequentist linear regression and – newly developed by us – Bayesian linear regression with normal distribution as likelihood and normal or vague prior distributions.

The approach is relatively robust towards changes in the prior distribution. Generally speaking, only very informative prior distributions that contradict the likelihood could pose serious problems due to the projection of the original data set onto the subspace. From a modelling perspective, such a situation would be undesirable and problematic, it is thus not a major limitation. While the generalisations to hierarchical models and to p-generalised normal distributions are not included in the theoretical guarantees, following this logic, they are included to some extent. The generalisation towards p-generalised normal distributions as likelihood is more problematic, here, the theoretical justification is still open.

The Merge & Reduce approach is also useful for very large data sets with $n \gg p$, but the focus does not lie on a reduction of the number of observations. Instead, the aim is to split the data set, conduct the analysis on smaller blocks and combine the resulting models efficiently.

This avoids memory problems with large data sets. When done serially, the overhead required for the technique only logarithmically depends on n. However, due to its nature, Merge & Reduce can also easily be parallelised, making it possible to utilise the advantages of current computing clusters.

References

Pankaj K. Agarwal, Sariel Har-Peled, and Kasturi R. Varadarajan. Approximating extent measures of points. *Journal of the ACM*, **51**(4):606–635, 2004. 55

Nir Ailon and Edo Liberty. Fast dimension reduction using Rademacher series on dual BCH codes. *Discrete & Computational Geometry*, **42**(4):615–630, 2009. 20

Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *Journal of Computer and System Sciences*, **58**(1):137–147, 1999. 19

Suhrid Balakrishnan and David Madigan. A one-pass sequential Monte Carlo method for Bayesian analysis of massive datasets. *Bayesian Analysis*, **1**(2):345–361, 2006. 15

Anjishnu Banerjee, David B. Dunson, and Surya T. Tokdar. Efficient Gaussian process regression for large datasets. *Biometrika*, **100**(1):75–89, 2013. 14

Richard Baraniuk, Mark Davenport, Ronald Devore, and Michael Wakin. A simple proof of the restricted isometry property for random matrices. *Constructive Approximation*, **28**(3), 2007. 14

Rémi Bardenet, Arnaud Doucet, and Chris Holmes. Towards scaling up Markov chain Monte Carlo: an adaptive subsampling approach. In *International Conference on Machine Learning*, pages 405–413, 2014. 16

Mark A. Beaumont, Wenyang Zhang, and David J. Balding. Approximate Bayesian Computation in population genetics. *Genetics*, **162**(4):2025–2035, 2002. 16

Austin R. Benson, David F. Gleich, and James Demmel. Direct QR factorizations for tall-and-skinny matrices in MapReduce architectures. In *Proceedings of the 2013 IEEE International Conference on Big Data*, pages 264–272, 2013. 14

Jon Louis Bentley and James B. Saxe. Decomposable searching problems I: Static-to-dynamic transformation. *Journal of Algorithms*, **1**(4):301–358, 1980. 54

William M. Bolstad. *Understanding Computational Bayesian Statistics.* Wiley series in computational statistics. Wiley, 2010. 8, 108

Björn Bornkamp, Arno Fritsch, Oliver Kuss, and Katja Ickstadt. Penalty specialists among goalkeepers: A nonparametric Bayesian analysis of 44 years of German Bundesliga. In *Statistical Inference, Econometric Analysis and Matrix Algebra: Festschrift in honour of Götz Trenkler*, pages 63–76. Springer, 2009. 94

Christos Boutsidis and Alex Gittens. Improved matrix algorithms via the Subsampled Randomized Hadamard Transform. *SIAM Journal on Matrix Analysis and Applications*, **34**(3): 1301–1340, 2013. 20

Christos Boutsidis and Malik Magdon-Ismail. Faster SVD-truncated regularized least-squares. In *Proceedings of the 2014 IEEE International Symposium on Information Theory*, pages 1321–1325, 2014. 14

Christos Boutsidis, Anastasios Zouzias, and Petros Drineas. Random projections for k-means clustering. In *Proceedings of the* 24^{th} *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 298–306, 2010. 14

Nicolas Bruno and Surajit Chaudhuri. Physical design refinement: The 'merge-reduce' approach. *ACM Transactions on Database Systems*, **32**(4):28, 2007. 55

Emmanuel J. Candès, Justin K. Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, **52**(2):489–509, 2006. 14

Samprit Chatterjee and Ali S. Hadi. *Sensitivity analysis in linear regression.* Wiley series in probability and mathematical statistics. Wiley, 1988. 5

Kenneth L. Clarkson. Subgradient and sampling algorithms for ℓ_1 regression. In *Proceedings of the* 16^{th} *annual ACM-SIAM symposium on Discrete algorithms (SODA)*, pages 257–266. Society for Industrial and Applied Mathematics, 2005. 55

Kenneth L. Clarkson and David P. Woodruff. Numerical linear algebra in the streaming model. In *Proceedings of the* 41^{st} *Annual ACM Symposium on Theory of Computing (STOC)*, pages 205–214, 2009. 14, 19

Kenneth L. Clarkson and David P. Woodruff. Low rank approximation and regression in input sparsity time. In *Proceedings of the* 45^{th} *Annual ACM Symposium on Theory of Computing (STOC)*, pages 81–90, 2013. 20

Kenneth L. Clarkson and David P. Woodruff. Sketching for M-estimators: A unified approach to robust regression. In *Proceedings of the* 26^{th} *Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 921–939. Society for Industrial and Applied Mathematics, 2015a. 55

Kenneth L. Clarkson and David P. Woodruff. Input sparsity and hardness for robust subspace approximation. In *Proceedings of the* 56^{th} *Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 310–329, 2015b. 55

Kenneth L. Clarkson, Petros Drineas, Malik Magdon-Ismail, Michael W. Mahoney, Xiangrui Meng, and David P. Woodruff. The fast Cauchy transform and faster robust linear regression. *SIAM Journal on Computing*, **45**(3):763–810, 2016. 55

Michael B. Cohen, Sam Elder, Cameron Musco, Christopher Musco, and Madalina Persu. Dimensionality reduction for k-means clustering and low rank approximation. In *Proceedings of the* 47^{th} *Annual ACM Symposium on Theory of Computing (STOC)*, 2015a. 14

Michael B. Cohen, Yin Tat Lee, Cameron Musco, Christopher Musco, Richard Peng, and Aaron Sidford. Uniform sampling for matrix approximation. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science (ITCS)*, pages 181–190, 2015b. 55

Paul G. Constantine and David F. Gleich. Tall and skinny QR factorizations in MapReduce architectures. In *Proceedings of the* 2^{nd} *International Workshop on MapReduce and Its Applications*, pages 43–50. ACM, 2011. 14

R. Dennis Cook. Detection of influential observation in linear regression. *Technometrics*, **19**(1): 15–18, 1977. 6

Katalin Csilléry, Michael G.B. Blum, Oscar E. Gaggiotti, and Olivier François. Approximate Bayesian Computation (ABC) in practice. *Trends in Ecology and Evolution*, **25**(7):410–418, 2010. 16

Anirban Dasgupta, Petros Drineas, Boulos Harb, Ravi Kumar, and Michael W. Mahoney. Sampling algorithms and coresets for ℓ_p regression. *SIAM Journal on Computing*, **38**(5):2060–2078, 2009. 55

James Demmel, Laura Grigori, Mark Hoemmen, and Julien Langou. Communication-optimal parallel and sequential QR and LU factorizations. *SIAM Journal on Scientific Computing*, **34** (1), 2012. 14, 15

Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`. 33

David L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, **52**(2): 1289–1306, 2006. 14

Matt Dowle and Arun Srinivasan. *data.table: Extension of 'data.frame'*, 2017. URL `https://CRAN.R-project.org/package=data.table`. R-package version 1.10.4-3. 26

Petros Drineas, Michael W. Mahoney, and S. Muthukrishnan. Sampling algorithms for ℓ_2 regression and applications. In *Proceedings of the* 17^{th} *Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1127–1136. Society for Industrial and Applied Mathematics, 2006. 55

Petros Drineas, Michael W. Mahoney, and S. Muthukrishnan. Relative-error CUR matrix decompositions. *SIAM Journal on Matrix Analysis and Applications*, **30**(2):844–881, 2008. 55

William DuMouchel, Chris Volinsky, Theodore Johnson, Corinna Cortes, and Daryl Pregibon. Squashing flat files flatter. In *Proceedings of the* 5^{th} *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 6–15, 1999. 13

Hadi Fanaee-T and João Gama. Event labeling combining ensemble detectors and background knowledge. *Progress in AI*, **2**(2-3):113–127, 2014. 33

Dan Feldman, Matthew Faulkner, and Andreas Krause. Scalable training of mixture models via coresets. In *Proceedings of the* 25^{th} *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 2142–2150, 2011. 93

Dan Feldman, Melanie Schmidt, and Christian Sohler. Turning big data into tiny data: Constant-size coresets for k-means, PCA and projective clustering. In *Proceedings of the* 24^{th} *Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1434–1453. Society for Industrial and Applied Mathematics, 2013. 13

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The Elements of Statistical Learning: Data mining, Inference and Prediction.* Springer-Verlag, 2^{nd} edition, 2009. 13

Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis*. Texts in Statistical Science. CRC Press, 3^{rd} edition, 2014. 7, 39, 40, 108

Stuart Geman and Donald Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, **6**(6): 721–741, 1984. 110

Leo N. Geppert, Katja Ickstadt, Alexander Munteanu, Jens Quedenfeld, and Christian Sohler. Random projections for Bayesian regression. *Statistics and Computing*, **27**(1):79–101, 2017. 17, 18, 19, 20, 21, 22, 33, 95

Loïc Giraldi, Olivier P. Le Maître, Ibrahim Hoteit, and Omar M. Knio. Optimal projection of observations in a Bayesian setting. *Computational Statistics & Data Analysis*, **124**:252 – 276, 2018. 15

Peter Gnändinger. Modellierung der Elfmeterfähigkeiten von Torhütern und Schützen. Master's thesis, TU Dortmund, 2018. 94

Jürgen Groß. *Linear Regression*, volume **175** of *Lecture Notes in Statistics*. Springer-Verlag, 2003. 2, 3, 4, 5, 6, 61

Rajarshi Guhaniyogi and David Brian Dunson. Bayesian compressed regression. *Journal of the American Statistical Association*, **110**(512):1500–1514, 2015. 14

Sariel Har-Peled and Soham Mazumdar. On coresets for k-means and k-median clustering. In *Proceedings of the* 36^{th} *Annual ACM Symposium on Theory of Computing (STOC)*, pages 291–300, 2004. 55

Frank E. Harrell, Jr. *Regression Modeling Strategies*. Springer-Verlag, 2001. 71, 84

Martin Hilbert and Priscila López. The world's technological capacity to store, communicate, and compute information. *science*, page 1200970, 2011. 8

Matthew D. Hoffman and Andrew Gelman. The No-U-turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15(1):1593–1623, 2014. 24, 112, 113

Jonathan H. Huggins, Trevor Campbell, and Tamara Broderick. Coresets for scalable Bayesian logistic regression. In *Proceedings of the* 30^{th} *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 4080–4088, 2016. 55

Shihao Ji and Lawrence Carin. Bayesian compressive sensing and projection optimization. In *Proceedings of the* 24^{th} *International Conference on Machine Learning (ICML)*, pages 377–384, 2007. 14

William B. Johnson and Joram Lindenstrauss. Extensions of Lipschitz mappings into a Hilbert space. *Contemporary mathematics*, 26(189-206):1, 1984. 11

Ian T. Jolliffe. *Principal component analysis*. Springer-Verlag, 2^{nd} edition, 2002. 13

Michael Kerber and Sharath Raghvendra. Approximation and streaming algorithms for projective clustering via random projections. *CoRR*, abs/1407.2063, 2014. 14

Walter Krämer and Harald Sonnberger. *The Linear Regression Model under Test*. Physica-Verlag, 1986. 3, 60

Niels Lategahn. Vergleich von Methoden zur Auswahl von Beobachtungen bei Regression mit fehlenden Y-Werten. Master's thesis, TU Dortmund, 2016. 92, 114, 115

Jonathan Law and Darren J. Wilkinson. Composable models for online Bayesian analysis of streaming data. *Statistics and Computing*, 28(6):1119–1137, November 2018. 55

Mu Li, Gary L. Miller, and Richard Peng. Iterative row sampling. In *Proceedings of the* 54^{th} *Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 127–136, 2013. 55

Dennis V. Lindley and Adrian F.M. Smith. Bayes estimates for the linear model. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–41, 1972. 39, 40, 41

Ping Ma, Michael W. Mahoney, and Bin Yu. A statistical perspective on algorithmic leveraging. In *Proceedings of the* 31^{st} *International Conference on Machine Learning (ICML)*, pages 91–99, 2014. 14

Chris A. Mack. Fifty years of Moore's law. *IEEE Transactions on Semiconductor Manufacturing*, 24(2):202–207, 2011. 8

David Madigan, Nandini Raghavan, William DuMouchel, Martha Nason, Christian Posse, and Greg Ridgeway. Likelihood-based data squashing: A modeling approach to instance construction. *Data Mining and Knowledge Discovery*, 6(2):173–190, 2002. 13

Thiago G. Martins, Daniel Simpson, Finn Lindgren, and Håvard Rue. Bayesian computing with INLA: New features. *Computational Statistics and Data Analysis*, 67:68–83, 2013. 16

Peter McCullagh and John A. Nelder. *Generalized Linear Models.* Monographs on Statistics and Applied Probability. Chapman and Hall, 2^{nd} edition, 1989. xii, 4, 6, 61, 65, 66

Alejandro Molina, Alexander Munteanu, and Kristian Kersting. Core dependency networks. In *Proceedings of the* 32^{nd} *AAAI Conference on Artificial Intelligence*, 2018. 55

Douglas C. Montgomery and Elizabeth A. Peck. *Introduction to Linear Regression Analysis.* Wiley, 3^{rd} edition, 1992. 3, 60

Steffen Müller. Untersuchung von Regression auf eingebetteten Datensätzen unter Verwendung von verschiedenen Abstandsnormen und Penalisierungstermen. Master's thesis, TU Dortmund, 2016. 39, 44, 53, 92

Jonas Münch. Untersuchung von Möglichkeiten zur Darstellung der Positionsdaten im Fußball. Bachelor's thesis, TU Dortmund, 2016. 94

Alexander Munteanu. *On large-scale probabilistic and statistical data analysis.* PhD thesis, TU Dortmund, 2018. 17

Alexander Munteanu and Chris Schwiegelshohn. Coresets-methods and history: A theoreticians design pattern for approximation and streaming algorithms. *KI-Künstliche Intelligenz*, 32(1): 37–53, 2018. 55

Alexander Munteanu, Chris Schwiegelshohn, Christian Sohler, and David P. Woodruff. On coresets for logistic regression. *CoRR*, abs/1805.08571, 2018. 55

S. Muthukrishnan. Data streams: Algorithms and applications. *Found. Trends Theor. Comput. Sci.*, **1**(2), 2005. 25

Saralees Nadarajah. A generalized normal distribution. *Journal of Applied Statistics*, 32(7): 685–694, 2005. 42

Radford M. Neal. MCMC using Hamiltonian dynamics. In Steve Brooks, Andrew Gelman, Galin L. Jones, and Xiao-Li Meng, editors, *Handbook of Markov Chain Monte Carlo*, Handbooks of Modern Statistical Methods, pages 113–162. CRC Press, 2011. 111, 112

Jelani Nelson and Huy L. Nguyễn. OSNAP: Faster numerical linear algebra algorithms via sparser subspace embeddings. In *Proceedings of the* 54^{th} *Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 117–126, 2013a. 20

Jelani Nelson and Huy L. Nguyễn. Sparsity lower bounds for dimensionality reducing maps. In *Proceedings of the* 45^{th} *Annual ACM Symposium on Theory of Computing (STOC)*, pages 101–110, 2013b. 20

Jelani Nelson and Huy L. Nguyễn. Lower bounds for oblivious subspace embeddings. In *Proceedings of the* 41^{st} *International Colloquium on Automata, Languages, and Programming (ICALP), Part I*, pages 883–894, 2014. 19

Saurabh Paul, Christos Boutsidis, Malik Magdon-Ismail, and Petros Drineas. Random projections for linear support vector machines. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, **8**(4):22, 2014. 14

Jeff M. Phillips. Coresets and sketches. In *Handbook of Discrete and Computational Geometry*, pages 1269–1288. Chapman and Hall/CRC, 2017. 55

Matias Quiroz, Robert Kohn, Mattias Villani, and Minh-Ngoc Tran. Speeding up MCMC by efficient data subsampling. *Journal of the American Statistical Association*, 0(0):1–35, 2018a. URL `https://doi.org/10.1080/01621459.2018.1448827`. 16

Matias Quiroz, Minh-Ngoc Tran, Mattias Villani, and Robert Kohn. Speeding up MCMC by delayed acceptance and data subsampling. *Journal of Computational and Graphical Statistics*, 27(1):12–22, 2018b. 16

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2018. URL `https://www.R-project.org/`. 24

Garvesh Raskutti and Michael Mahoney. Statistical and algorithmic perspectives on randomized sketching for ordinary least-squares. In *Proceedings of the* 32^{nd} *International Conference on Machine Learning (ICML)*, pages 617–625, 2015. 14

Jonathan Rathjens. Hierarchische Bayes-Regression bei Einbettung großer Datensätze. Master's thesis, TU Dortmund, 2015. 39, 92

Sashank J. Reddi, Barnabás Póczos, and Alexander J. Smola. Communication efficient coresets for empirical loss minimization. In *Proceedings of the* 31^{st} *Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 752–761, 2015. 55

Håvard Rue, Sara Martino, and Nicolas Chopin. Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations (with discussion). *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, **71**:319–392, 2009. 16

Florin Rusu and Alin Dobra. Pseudo-random number generation for sketch-based estimations. *ACM Transactions on Database Systems*, **32**(2), 2007. 19, 20

Tamás Sarlós. Improved approximation algorithms for large matrices via random projections. In *Proceedings of the* 47^{th} *Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 143–152, 2006. 14, 19

Holger Schwender and Katja Ickstadt. Identification of SNP interactions using logic regression. *Biostatistics*, 9(1):187–198, 2008. 93

Holger Schwender, Ingo Ruczinski, and Katja Ickstadt. Testing SNPs and sets of SNPs for importance in association studies. *Biostat*, 12(1):18–32, 2011. 93

Christian Sohler and David P. Woodruff. Subspace embeddings for the L_1-norm with applications. In *Proceedings of the* 43^{th} *Annual ACM Symposium on Theory of Computing (STOC)*, pages 755–764, 2011. 55

Stan Development Team. RStan: the R interface to Stan, 2018. URL `http://mc-stan.org/`. R-package version 2.17.3. 8, 24, 113

M. Th. Subbotin. On the law of frequency of error. *Matematicheskii sbornik*, 31(2):296–301, 1923. 42

Elad Tolochinsky and Dan Feldman. Coresets for monotonic functions with applications to deep learning. *CoRR*, abs/1802.07382, 2018. 55

William N. Venables and Brian D. Ripley. *Modern Applied Statistics with S*. Springer, New York, fourth edition, 2002. 67

Cédric Villani. *Optimal transport: Old and new.* Grundlehren der mathematischen Wissenschaften. Springer-Verlag, Berlin, 2009. 17

Max Welling, Yee Whye Teh, Christophe Andrieu, Jakub Kominiarczuk, Ted Meeds, Babak Shahbaba, and Sebastian Vollmer. Bayesian inference & Big Data: A snapshot from a workshop. *ISBA Bulletin*, **21**(4):8–11, 2014. 9, 21

Alexander Wollenberg. Reduktion hochdimensionaler Datensätze für die logische Regression unter Verwendung von Leverage Scores mit besonderer Berücksichtigung von SNP-Daten. Master's thesis, TU Dortmund, 2016. 93

David P. Woodruff and Qin Zhang. Subspace embeddings and ℓ_p-regression using exponential random variables. In *Proceedings of the* 26^{th} *Annual Conference on Learning Theory (COLT)*, pages 546–567, 2013. 55

Jiyan Yang, Xiangrui Meng, and Michael W. Mahoney. Implementing randomized matrix algorithms in parallel and distributed environments. *CoRR*, abs/1502.03032, 2015. 14

Appendix A: Overview of MCMC-methods

«Get out!»– «I'm imparting my knowledge on the bro.»

(Hunt for the Wilderpeople)

In cases where Bayesian regression models cannot be solved analytically, approximations form a viable and useful alternative. MCMC-techniques currently form the gold standard for such approximations. They are reliable and can be employed for an extremely broad class of models. Additionally, it is possible to check for convergence, thus enabling to check whether the approximation is good or further iterations are needed. A disadvantage, however, is that MCMC-methods are not efficient. The running time required before convergence is reached may be substantial. In the following, I will briefly introduce common MCMC-methods, leading up to Hamiltonian Monte Carlo and the No-U-turn sampler. The latter is the only MCMC-method employed in this thesis.

Monte Carlo integration is a numerical integration technique with the basic idea of randomly drawing observations and evaluating them according to the function of interest. Rejection sampling presents one possibility of a Monte Carlo integration technique. Here, the idea is to sample a candidate value from an easily available proposal distribution and then to decide to either keep or reject this candidate based on how well it fits to the posterior distribution of interest. In more detail, let $\underline{\theta}_C$ be the candidate sampled from the proposal distribution $q(\underline{\theta})$, where $\underline{\theta}$ may be of one or more dimensions. Let $p(\underline{\beta}|X, \underline{Y})$ be the posterior distribution of interest and $p^*(\underline{\beta}|X, \underline{Y})$ the unnormalised posterior

$$p^*(\underline{\beta}|X, \underline{Y}) = L(\underline{\beta}|X, \underline{Y}) \times p(\underline{\beta}). \tag{1}$$

This means that $p(\underline{\beta}|X,\underline{Y}) = p^*(\underline{\beta}|X,\underline{Y})/\int p^*(\underline{\beta}|X,\underline{Y})d\underline{\beta}$. In the following equations we utilise the unnormalised posterior, but it is straightforward to replace it with the normalised posterior.

Rejection sampling decides whether to keep $\underline{\theta}_C$ or reject it based on

$$\alpha < \frac{p^*(\underline{\theta}_C)}{Mq(\underline{\theta}_C)}, \tag{2}$$

where $M > 0$ is a constant such that

$$\frac{p^*(\underline{\theta})}{q(\underline{\theta})} \leq M \text{ for all } \underline{\theta} \in \Theta \tag{3}$$

and α is a single draw from $\mathsf{Unif}(0,1)$. If a candidate value is accepted, it is added to the sample that represents the posterior distribution while the sample remains unchanged if it is rejected. In both cases, the next step is to sample a new candidate from the proposal distribution and continue until the desired number of observations are in the sample or a maximum number of iterations is reached.

The efficiency of rejection sampling depends on the relation between the posterior distribution $p(\underline{\theta})$ and the proposal distribution $q(\underline{\theta})$. If $q \propto p$, all candidates are well-fitting as draws from the posterior, leading to an acceptance rate of 1. If p and q become very different, the constant M needs to be set to a high number to fulfil the condition given in Equation (3). This leads to low acceptance rates, which means that longer sampling times are necessary to obtain the desired sample size. An upside to this is that it is straightforward to recognise whether rejection sampling works efficiently or not by monitoring the acceptance rates.

There are some other Monte Carlo approaches beside rejection sampling that are based on sampling directly from the posterior distribution or its unnormalised version. However, all of these methods tend to become inefficient when the prior distribution is non-informative compared to the likelihood, i.e. it covers a distinctly wider range of values, or as the number of variables and, with that, parameters grows (Bolstad, 2010).

For that reason, rejection sampling has in many situations been replaced by more efficient methods, some of which are introduced below. According to Gelman et al. (2014), rejection sampling is used "in some fast methods for sampling from standard univariate distributions" as well as "for generic truncated multivariate distributions, if the proportion of the density mass in the truncated part is not close to 1" (both in Gelman et al., 2014, p. 265).

Algorithm 3: Metropolis-Hastings algorithm

output MCMC-sample of size T
Set initial value $\underline{\theta}_1$
for $t \in \{2, \ldots, T\}$ **do**
  draw $\underline{\theta}_C$ from $q(\underline{\theta}|\underline{\theta}_{t-1})$
  calculate acceptance probability

$$\alpha = \min\left(1, \frac{p(\underline{\theta}_C|X, \underline{Y})q(\underline{\theta}_{t-1}|\underline{\theta}_C)}{p(\underline{\theta}_{t-1}|X, \underline{Y})q(\underline{\theta}_C|\underline{\theta}_{t-1})}\right) \tag{4}$$

  draw $u \sim \mathsf{Unif}(0, 1)$
  if $u < \alpha$ **then** $\underline{\theta}_t \leftarrow \underline{\theta}_C$
  else $\underline{\theta}_t \leftarrow \underline{\theta}_{t-1}$

Markov Chain Monte Carlo (MCMC) methods offer a more efficient alternative to direct sampling from the posterior distrubtion. MCMC-methods are currently the standard in Bayesian analysis. They are reliable and can handle a very broad range of models. However, depending on the model and the size of the data set, MCMC-methods can be very slow.

The idea behind MCMC-methods is the following: the desired posterior distribution is proportional to the product of the prior distribution and the likelihood. This product is divided by the result of a high-dimensional integration, which is hard to compute. MCMC-algorithms start with initial values for all parameters and in every iteration sample a new candidate from a proposal distribution. This new candidate may only depend on the current value, but not on values before that. This is known as Markov property. The algorithm then checks whether the new candidate fits well with the posterior distribution by considering the posterior distribution of the candidate divided by the posterior distribution of the current value. By construction, the high-dimensional integral cancels out in the fraction, making it possible to decide whether to keep the candidate value or not based on this fraction.

The Metropolis-Hastings (MH) algorithm offers one possibility of obtaining a suitable Markov chain. Let $\underline{\theta}_1$ be the starting value for the Markov chain and let $\underline{\theta}_t, t = 2, \ldots, T$, be further elements of the Markov chain, where T is the desired length of the chain. The elements $\underline{\theta}_t$ of the Markov chain are iteratively chosen following the steps given in Algorithm 3. We will discuss some things to consider when choosing T later on.

Although the posterior distribution features prominently in Equation (4), the MH algorithm only requires knowledge of the unnormalised posterior distribution. The constant term that normalises the distribution is identical in enumerator and denominator and thus cancels out. In

Equation (4), only the current state of the Markov Chain and the new candidate value are taken into account, the MH algorithm has the Markov property.

The stationary distribution of the Markov chain constructed by the MH algorithm 3 is the desired posterior distribution $p(\underline{\beta}|X, \underline{Y})$. The choice of the proposal distribution q does not influence the stationary distribution, however, it may influence the acceptance rate and the speed of convergence. In general, acceptance rates between 0.3 and 0.7 are desirable. Higher rates may indicate that the variance of the proposal distribution is low relative to the variance in the posterior distribution. This may lead to slow exploration of regions with relatively high probability mass, in such cases, the posterior distribution may not be characterised fully by the resulting MCMC-sample. If on the other hand acceptance rates are very low, this indicates a high variance of the proposal distribution compared with the posterior's variance. In this case, the MCMC-sample contains the same observations multiple times, reducing the effective sample size.

Blockwise Metropolis-Hastings and Gibbs sampling Algorithm 3 may be used for a single parameter or for multiple parameters. For the multi-parameter case, blockwise Metropolis-Hastings is an important special case of the MH algorithm. In the MH algorithm, one p-dimensional candidate vector is drawn and either accepted or rejected. In the blockwise version, the parameters are partitioned into J blocks. If variables are known to be correlated they may be put into the same block, but this is not a necessity.

When the parameters have been partitioned, we replace drawing a single parameter value $\underline{\theta}_C$ by iteratively drawing a candidate value for each of the J blocks, i.e. $\underline{\theta}_{C,j},\ j = 1, \ldots, J$. The proposal distribution now depends on the current value of the Markov chain for the parameters in block j and on the current values for the other blocks. Because the sampling is done iteratively, when the candidate value for entry t of block j is drawn, the candidate depends on the t^{th} value for blocks $j^* < j$ and on the $(t-1)^{\text{st}}$ value for blocks $j' > j$.

Gibbs sampling is a special case of the blockwise MH algorithm. It was developed by Geman and Geman (1984). To construct a Gibbs sampler, the proposal distribution for block j is chosen as full conditional distribution given the values in all other blocks. By this choice, all terms in the acceptance probability cancel out, leaving 1, i.e. every candidate value is accepted with probability 1. This high acceptance rate is possible because every proposal distribution goes well together with the respective posterior distribution. Gibbs sampling is very efficient, but it requires knowledge of the full conditional distributions, which is often not the case.

Random walk Metropolis-Hastings For both the basic MH algorithm as well as the blockwise MH algorithm, the proposal distribution may also be chosen symmetrically, such that $q(\underline{\theta}_{t-1}|\underline{\theta}_C) = q(\underline{\theta}_C|\underline{\theta}_{t-1})$. In such cases, the proposal cancels out of Equation (4), leaving

$$\alpha = \min\left(1, \frac{p(\underline{\theta}_C|X, \underline{Y})}{p(\underline{\theta}_{t-1}|X, \underline{Y})}\right) \tag{5}$$

in the basic version. When there are multiple parameters (basic or blockwise version), the proposal distribution is symmetrical in every component. This approach is also called random walk MH algorithm, because the next candidate does not depend on the current position of the Markov chain. From Equation (5) we can see that if the candidate value is an improvement over the current value in terms of the posterior distribution, it is always accepted. If the candidate value is further away from the centre of the posterior distribution, it may still be accepted depending on how much lower the probability mass is. This ensures that the algorithm generally moves in the direction of high probability mass, but also explores the posterior distribution. This is important as the aim is not optimisation, but characterisation of the posterior distribution.

Generally speaking, the variance of q controls the average step size. Similarly to our earlier remarks about the proposal distribution's variance, in the case of a random walk MH, the variance of q influences the efficiency of the algorithm and the number of iterations required to obtain a well-fitting sample of the posterior distribution.

Hamiltonian Monte Carlo The first step in every MCMC-algorithm presented here is to draw an initial value $\underline{\theta}_1$. This initial value may by chance be close to the regions with high probability mass, but – especially in higher-dimensional cases – it more likely will be some distance away.

The MCMC-algorithms introduced so far employ the same proposal distribution for every iteration t, $t = 1, \ldots, T$, resulting in constant step sizes. This clearly is not optimal in cases where the algorithm first has to cover some distance to reach the region with high probability mass and then needs to explore this region. Adaptive step sizes provide a remedy against this, but require additional calculations to obtain useful values.

Hamiltonian Monte Carlo (HMC) provides a solution for the changing step sizes. HMC is also called hybrid Monte Carlo as it combines MCMC-techniques with a deterministic part which is influenced by the physical concept of Hamiltonian dynamics. For every component θ_j of $\underline{\theta}$, a momentum variable ϕ_j is introduced. Neal (2011) compares the two components to a puck that slides over ice – or, more generally, a frictionless surface – with varying heights. The puck has

a position $\underline{\theta}$ and a momentum $\underline{\phi}$. On a plane section, the momentum will remain unchanged by and large, whereas when going up a slope, the puck will be carried in the same direction by the momentum, which in turn will be slowed down. This way, position and momentum have an influence on each other.

As an MCMC-technique, HMC replaces the updating algorithm by the following steps (Hoffman and Gelman, 2014; Neal, 2011). First, a new value for the momentum is drawn from a multivariate normal distribution $N(0, I_p)$, where I_p is the p-dimensional unity matrix. In the second step, both the position $\underline{\theta}$ and the momentum $\underline{\phi}$ are updated simultaneously. To that end, Hamiltonian dynamics are simulated using L steps of the leapfrog algorithm. The leapfrog algorithm is employed to discretise the continuous Hamiltonian dynamics. Every leapfrog step consists of two updates in three steps, a full update of the position $\underline{\theta}$ as second step, preceded and followed by half an update of the momentum, where the newest value of $\underline{\theta}$ is employed. The updated value of $\underline{\theta}$ is obtained by adding an ι-fraction of the momentum to the current value of $\underline{\theta}$. For the half-updates of the momentum, an $\iota/2$-fraction of the product of the gradient $\underline{\Delta}_{\underline{\theta}}$ with respect to $\underline{\theta}$ and the logarithm $\ln p(\underline{\theta}|\underline{Y}, X)$ of the current parameter value is added to the current momentum. The parameter ι is also called the stepsize. After L of such steps, we draw $u \sim \mathsf{Unif}(0, 1)$ and accept the new values if

$$u < \alpha = min\left(1, \frac{\exp(\ln p(\underline{\theta}^*|\underline{Y}, X) - \frac{1}{2}\underline{\phi}^*\underline{\phi}^*)}{\exp(\ln p(\underline{\theta}_{t-1}|\underline{Y}, X) - \frac{1}{2}\underline{\phi}_{t-1}\underline{\phi}_{t-1})}\right). \tag{6}$$

When the candidate values are rejected, $\underline{\theta}_t = \underline{\theta}_{t-1}$ and $\underline{\phi}_t = \underline{\phi}_{t-1}$. When the candidate values are accepted, $\theta_t = \theta^*$ and $\underline{\phi}_t = -\underline{\phi}^*$. $\underline{\phi}^*$ is negated to make the proposal symmetric and obtain a time-reversible Markov chain. Since the proposal is time-reversible and the leapfrog algorithm preserves volumes, the HMC algorithm is a valid Metropolis proposal. For more details on Hamiltonian dynamics, the HMC algorithm and its properties as well as extensions and variations, please refer to Neal (2011).

The HMC approach works more efficiently in many situations, but requires tuning of two vital parameters, the number of leapfrog steps L and the stepsize ι. As Hoffman and Gelman (2014) note, choosing ι too low will result in very small steps while a value that is chosen too large will result in an inaccurate simulation of Hamiltonian dynamics and low acceptance rates. Similarly, if too few leapfrog steps L are conducted, successive samples are too close to one another, but if L is too large, the algorithm may loop back and retrace its own steps. All of this results in undesirable inefficient behaviour.

To avoid tuning these parameters, (Hoffman and Gelman, 2014) propose their No-U-turn sampler (NUTS), which requires no parameter tuning while achieving results similar or better compared to a well-tuned HMC-algorithm, according to the authors. The Stan development team's homepage[1] offers documentation of the Stan modelling language as well as help for implementations of Stan for different statistical software like R (Stan Development Team, 2018), Python, Julia, MATLAB or command-line terminals[2].

They follow different approaches for the two parameters. For the stepsize ι, Hoffman and Gelman (2014) propose stochastic optimisation with vanishing adaptation, which can be used for both HMC and NUTS. To automatically tune L, Hoffman and Gelman (2014) employ a doubling algorithm. Whenever a new candidate is sampled, the NUTS algorithm builds a set of possible candidate values in multiple steps. Initially, only the current position and momentum are added to the set. In the first step, one new candidate value (with both position and momentum) is added, followed by two candidates in the second step and 2^{j-1} candidates in the j^{th} step. This means that the total number of candidates after j steps is 2^j. In each step, the 2^{j-1} candidates can go either forwards or backwards from the respective outermost candidate in the direction of the current momentum. The process is stopped when new candidate values start retracing their own steps, thus suggesting candidates that have already been explored during the current search for a candidate or when a suggested candidate is associated with a very low probability of occurring.

In a basic version, the new position is determined by uniformly sampling one candidate from the set. As the set always includes the current position, the current state may remain unchanged. In a more efficient version, NUTS proposes one of the 2^{j-1} newly added candidates as new position in step j. According to Hoffman and Gelman (2014), this results in longer jumps on average as candidate state with low probability in the later stages of finding a new position can be compensated by already having chosen a new position in an earlier stage. Thus, small jumps are conducted with high probability while still allowing for longer jumps.

[1] https://mc-stan.org
[2] Full list available at https://mc-stan.org/users/interfaces/

Appendix B: Deviations from and violations of the assumptions

No eres tu, no eres tu, no eres tu, soy yo.

Luis Fonsi y Demi Lovato – Échame la culpa

Generally speaking, reduction of the number of observations in a regression context is easy as long as all observations are similar and follow the same, correctly specified model. In such a case, the influence of the selection method (including simple random sampling) on the results is low. This changes as soon as some observations are more important for the outcome model than others.

Lategahn (2016) introduces and analyses a number of scenarios designed to alter the importance of different observations. Some of the changes introduced represent violations from the assumptions of a linear regression model. The first two scenarios represent data sets where the observations are close together. To that end, Lategahn chooses an interval $[a_i, b_i] \subset [-3, 3]$ such that every column vector $\underline{X}_j \in [a_j, b_j]$ for $j \in \{1, \ldots, p\}$. Then, values of $\beta_j, j \in \{0, \ldots, p\}$ are generated. The values of Y are then obtained by multiplying the design matrix and β and adding a normally distributed error term $\underline{\eta}$. In the first case, the variance of the error term is chosen to be 1, in the second case it is set to 100.

In addition to these non-problematic cases, Lategahn also considers violations of the assumptions. In detail, these are the following.

1. Outliers in the values of X: A fraction of 5% of the data is generated from a uniform distribution $U(b_i + 9.75, b_i + 10.25)$. The values of $\underline{Y}$ follow the model as before with an error term variance of 100.

2. Heteroscedasticity: $\underline{\eta} \sim N(0, \sigma_i^2)$ where $\sigma_i^2 = 1 + 99\frac{y_i - \min(y)}{\max(y) - \min(y)}$. This leads to a smaller values of $\underline{y}$ being systematically closer to the hyperplane than larger values of $\underline{y}$.

3. Outliers in the values of $\underline{Y}$: With $\sigma^2 = 1$, the realisation of a $N(20, 1)$ distributed random variable is added to y_i for a fraction of 5% of all observations. Here, he differentiates between two cases: in the first case, the outliers are chosen randomly from all observations. Thus, they tend to be equally spread along the range of $\underline{y}$. In the second case, he randomly draws a value in the range of X and concentrate all outliers in this region. This region may additionally contain non-outlying observations.

4. Data that come from different populations: different fractions of the data follow different regression models. This is not taken into account when modelling, one common regression model for both populations is built.

5. Unobserved transformation: The values of $\underline{y}$ are generated using a sine-transformation for all columns $\underline{X}_i, i \in \{1, \ldots, p\}$. The transformation is not taken into account in the model. Since the generated data are enclosed by $[-3, 3] \approx [-\pi, \pi]$, the data cover almost one period of the sine function.

6. Interactions: between five and ten interactions are chosen at random. A variable may "interact" with itself, resulting in a quadratic term. None of this is taken into account in the model.

In some of the situations a linear model is not the optimal choice, it may even be inadequate. However, the aim here is to study the effects the above-mentioned violations have on the linear model and especially on the reduced data set. With the exception of outliers in the values of X, the other violations cannot easily be spotted before the analysis as the assumptions in Lategahn (2016) is that values of $\underline{Y}$ are not known.

Broadly speaking, Lategahn (2016) finds that methods, which combine exploration and sampling according to importance measures, work best over all scenarios considered. Sampling according to, e.g., leverage scores alone is not sufficient, but in combination with a method that ensures that all areas are represented in the sample it leads to good results.